MINISTRY OF EDUCATION AND TRAINING

TRƯỜNG ĐẠI HỌC FPT

FPT UNIVERSITY

Capstone Project Document

Surveillance Camera

Group – Embedded System	
Group Members	Nguyễn Tư Duy – Team Leader - SE60775 Hoàng Minh Thắng – Team Member – SE60754 Đỗ Văn Thanh Tú – Team Member – SE60500 Vũ Quế Anh – Team Member – 00840
Supervisor	Mr. Trương Vĩnh Lân
Ext Supervisor	Mr. Phan Duy Hùng
Capstone Project Code	SC

Ha Noi, January 2015

This page is intentionally left blank

ACKNOWLEDGEMENTS

We couldn't have finished this project without the help and technical assistance of many people. The amount of time and effort that goes into this project is hard to measure.

Especially, we truly had a fantastic person helping us - Mr. Phan Duy Hung, his professional guidance and his experience help us to overcome the obstacles, even the hardest time when we think we can't continue with this project anymore. He has truly had a significant impact on the success of this project.

We would like to extend a special thank you to our friends who have support us throughout 4 months of this project.

Table of Contents

Table of Contents 0

List of Tables 6

List of Figures 7

Definitions, Acronyms, and Abbreviations 10

1. INTRODUCTION 12

1.1 Project Information 12

1.2 Introduction 12

1.3 Existing types of camera 15

1.4 Proposed Solution 16

1.5 Feature Functions 17

1.6 Project Team 17

2. SYSTEM PROJECT MANAGEMENT PLAN (SPMP) 18

2.1 Problem Definition 18

2.1.1 Name and Main Points of Capstone Project 18

2.1.2 Problem Abstract 18

2.1.3 Project Overview 18

2.2 Overview 20

2.2.1 System Process Model 20

2.2.2 Roles and Responsibilities 22

2.2.3 Tools and Techniques 23

2.3 Project Management Plan 25

2.3.1 Iteration 25
2.3.2 Iteration detail 27
2.3.3 Task Sheet: Assignments and Timetable 29
2.4 Convention Rules 29
3. SYSTEM REQUIREMENTS SPECIFICATIONS (SRS) 30
3.1 User Requirement Specification 30
3.2 System Requirement Specification 30
3.2.1 External Interface Requirements 30
3.2.2 Functional Requirements 31
3.2.3 Non-functional Requirements 56
4. SOFTWARE DESIGN DESCRIPTION (SDD) 57
4.1 Design Overview 57
4.2 System Architecture Design 57
4.3 Component Diagram 58
4.4 Detailed Design of Components 59
4.4.1 Hardware 59
4.4.2 Software 66
4.5 Flow Chart 74
4.5.1 Main Flow 74
4.5.2 InitCamera 75
4.5.3 Send Package 76
4.5.4 SmartConfig 77
4.6 Sequence Diagram 78

4.6.1 View Camera 78
4.6.2 SmartConfig 78
4.7 User Interface Design/Hardware Interface Design 79
4.7.1 Main Form 79
4.7.2 Settings Form 80
4.7.3 Playback Form 83
4.7.4 SmartConfig 84
4.8 Camera Database Design 86
5. SYSTEM IMPLEMENTATION & TEST (SIT) 87
5.1 Introduction 87
5.1.1 System Overview 87
5.1.2 Test Approach 87
5.1.3 Test plan 87
5.1.4 Test pass/fail criteria 87
5.1.5 Test environment 88
5.2 Test Case 89
5.2.1 Surveillance Camera (SC) Test Cases 89
5.2.2 SmartConfig Android App (SAA) Test Cases 92
5.2.3 Surveillance Camera Monitor (SCM) Test Cases 94
5.2.4 Test Report 101
5.3 Test Evident 101
5.2.1 Test SmartConfig Android App (SAA) 101
5.2.2 Test Surveillance Camera (SC) 103

5.2.3 Test Surveillance Camera Monitor (SCM) 104

6. SYSTEM USER'S MANUAL (SUM) 107

6.1 System Requirement 107

6.2 SmartConfig Android App (SAA) 107

6.3 Surveillance Camera Monitor (SCM) 110

6.3.1 Watching Video 110

6.3.2 Control Camera 112

6.3.3 Playback 120

7. REFERENCES 124

List of Tables

Table 1 – Project team 17
Table 2 – Roles and Responsibilities 22
Table 3 – Project Iterations 26
Table 4 – Phase 1- Preliminary study and planning 27
Table 5 – Phase 2 - Design, prepare hardware 27
Table 6 – Phase 3 - Develop camera firmware 28
Table 7 – Phase 4 - Develop window application 28
Table 8 – Phase 5 - Integration and system test for the behavior in System 29
Table 9 – Connect cameras to network use case Description 33
Table 10 – Watch Videos Use case Description 35
Table 11 – Add Camera Use Case Description 39
Table 12 – Change Name Use Case Description 42
Table 13 – Change Port Use Case Description 46
Table 14 – Change Description Use Case Description 48
Table 15 – Delete Camera Use Case Description 52
Table 16 – View Recorded Videos Use Case Description 55
Table 17 – SC Monitor Main Form Description 79
Table 18 – Setting Form Description 80
Table 19 – Add Camera Form Description 81
Table 20 – Edit Camera Form Description 82
Table 21 – Playback Form Description 83
Table 22 – SAA SmartConfig Tab Description 85
Table 23 –SAA Devices Tab Description 86
Table 24 – Test case for SC 89
Table 25 – Test case for SAA 92
Table 26 –Test Cases For SCM 95
Table 27 –Test Report 101
Table 28 – System Requirement 107

List of Figures

Figure 1 – Camera applications in real life 14
Figure 2 – Analog camera and how it works 15
Figure 3 – Analog camera and how it works 15
Figure 4 – Surveillance Camera System 16
Figure 5 – Boundaries of the System 20
Figure 6 – Iterative and Incremental development model 20
Figure 7 – Use case Diagram 31
Figure 8 –Connect cameras to network use case Diagram 32
Figure 9 – Watch Videos Use case Diagram 34
Figure 10 – Control Cameras Use case Diagram 37
Figure 11 – Add Camera Use case Diagram 38
Figure 12 – Change Name Use Case Diagram 41
Figure 13 – Change Port Use Case Diagram 44
Figure 14 – Change Description Use Case Diagram 47
Figure 15 – Delete Camera Use Case Diagram 50
Figure 16 – View Recorded Use Case Diagram 53
Figure 17 – System Architecture Design 57
Figure 18 – Component Diagram 58
Figure 19 – CC3200 Launchpad Overview 59
Figure 20 – CC3200 Launchpad Schematic 60
Figure 21 – CC3200 Overview of Peripherals 62
Figure 22 – MT9D111 Camera Module 62
Figure 23 – MT9D111 PIN 63
Figure 24 – MT9D111 Schematic 64
Figure 25 – Mapping Launchpad to Mt9d111 64
Figure 26 – SC Server Class Diagram 66
Figure 27 – Surveillance Camera Class Diagram 69
Figure 28 – Main Flowchart 74
Figure 29 – Init Camera Flowchart 75
Figure 30 – Send Package Flowchart 76
Figure 31 – SmartConfig Flowchart 77
Figure 32 – View Camera Sequence Diagram 78

Figure 33 – SmartConfig Sequence Diagram 78
Figure 34 – SC Monitor Main Form 79
Figure 35 – Setting Form 80
Figure 36 – Add Camera Form 81
Figure 37 – Edit Camera Form 82
Figure 38 – Playback Form 83
Figure 39 –SAA SmartConfig Tab 84
Figure 40 –SAA Devices Tab 85
Figure 41 – Camera Database Design 64
Figure 42 – SAA Searching 102
Figure 43 – SAA Show Result 102
Figure 44 – SC Connect Successful 103
Figure 45 – SC Send Data 104
Figure 46 – SCM Test ViewVideo 105
Figure 47 – SCM Test Playback 106
Figure 40 – SAA Welcome Screen 107
Figure 41 – Start connect camera to the network 108
Figure 42 – Searching for matched cameras 109
Figure 43 – Connect camera to the network succesful 110
Figure 44 – Before start watching 111
Figure 45 – Start watching 111
Figure 46 – Stop watching 112
Figure 47 – Setting screen 112
Figure 48 – Choose Add button 113
Figure 49 – Adding Screen 113
Figure 50 – Enter camera information 114
Figure 51 – Add succesful 114
Figure 52 – Return Setting screen 115
Figure 53 – Choose Camera for editting 116
Figure 54 – Editting screen 116
Figure 55 – Change camera information 117
Figure 56 – Edit successful 117
Figure 57 – Return Setting screen 118

Figure 58 – Choose camera for deleting 118
Figure 59 – Delete dialog appear 119
Figure 60 – Delete succesful 119
Figure 61 – Return Setting screen 120
Figure 62 – Choose Playback button 120
Figure 63 – Playback screen 121
Figure 64 – Hour buttons 121
Figure 65 – Video appear 122
Figure 66 – Play video 123

Definitions, Acronyms, and Abbreviations

IP	Internet Protocol	
SC	Surveillance Camera	The camera which is intended to design and implement,
SCM	Surveillance Camera Monitor	The window application to monitor and control the cameras
LAN	Local Area Network	
WAN	Wide Area Network	
GUI	Graphical user interface	
I2C	Inter-Integrated Circuit	I2C is a multi-master serial single-ended computer bus invented by the Philips semiconductor division.
I/O	Input/Output	
UART	Universal asynchronous receiver/transmitter	UART is a piece of computer hardware that translates data between parallel and serial forms.
MCU	Microcontroller unit	
PC	Personal Computer	
CMOS	Complementary Metal-Oxide-Semiconductor	
CCD	Charge Coupled Device	
CCTV	Closed-circuit television	
NTSC	National Television System Committee	
PAL	Phase Alternating Line	
SECAM	Sequential Color with Memory	
WPA	Wi-Fi Protected Access	A Wi-Fi standart

TKIP	Temporal Key Integrity Protocol	A Wi-Fi standart
AES	Advanced Encryption Standard	A Wi-Fi standart
SAA	SmartConfig Android App	Android application for configurating the network for cameras
DVR	Digital Video Recorder	
IoT	Internet of Things	
IFP	Image Flow Processor	
GPIO	General Purpose Input Output	
JTAG	Joint Test Action Group	
SWD	Serial Wire Debug	
ADC	Analog to Digital Converter	
JPEG	Joint Photographic Experts Group	An Image standard
TCP	Transmission Control Protocol	
HTTP	Hyper Text Transfer Protocol	
HTML	Hyper Text Markup Language	
DCT	Discrete Cosine Transform	

1. INTRODUCTION

1.1 Project Information

- Project name: **Surveillance Camera**
- Project Code: **SC**
- Product Type: **Camera , Embedded System, Windows Application**
- Start Date: **January, 2015**
- End Date: **April, 2015**

1.2 Introduction

Compared to the past, cameras are used more common all around the world nowadays due to the decreasing of the cost as well as the raise in the needs of a monitoring tool. There are a wide range of camera's uses and the main use is monitoring, people tend to use cameras to monitor the security of their houses, families, offices and poverties. For example, now many families can take care of their babies easier by putting cameras in baby's rooms; bosses can just sit in their rooms and manage their employees through cameras in offices. Modern cameras also provide many valuable features such as two-way communication - it is not only capable of allowing you to hear what is happening at the place where the camera is set up, but will also allow you to speak through the camera's built in speaker. The integrated sensors also make the cameras become more helpful and enlarge the list of uses of cameras, to be more specific, thermal and movement sensor help people know if there are thieves broke into their houses or buildings for example.

An Internet protocol camera, or IP camera, is a type of digital video camera commonly employed for surveillance, it also can connects to an IP network such as a LAN (Local Area Network) or the Internet. An IP camera can have either a CMOS or a CCD sensor, and is available in the same styles as traditional surveillance cameras such as Pan/Tilt/Zoom, domes, bullets, box, infrared, covert, and wireless.

IP cameras are typically equipped with an embedded web server and can be accessed and controlled over any IP network such as a WAN, LAN, Intranet, or Internet. By utilizing a standard web browser or client software users can view an

IP camera's video output from any local or remote location. IP cameras combine the capabilities of a camera with some PC functionality, do not require a direct connection to a PC to operate, and can be placed anywhere within a network. Just like any other PC on the network, an IP camera is a "network appliance". It has its own IP address, connects directly to a wired or wireless network and requires maintenance.

Through the network, the camera can stream video that can be accessed easily, either by using web browsers or apps on technology devices such as laptops, smart phones and tablets.

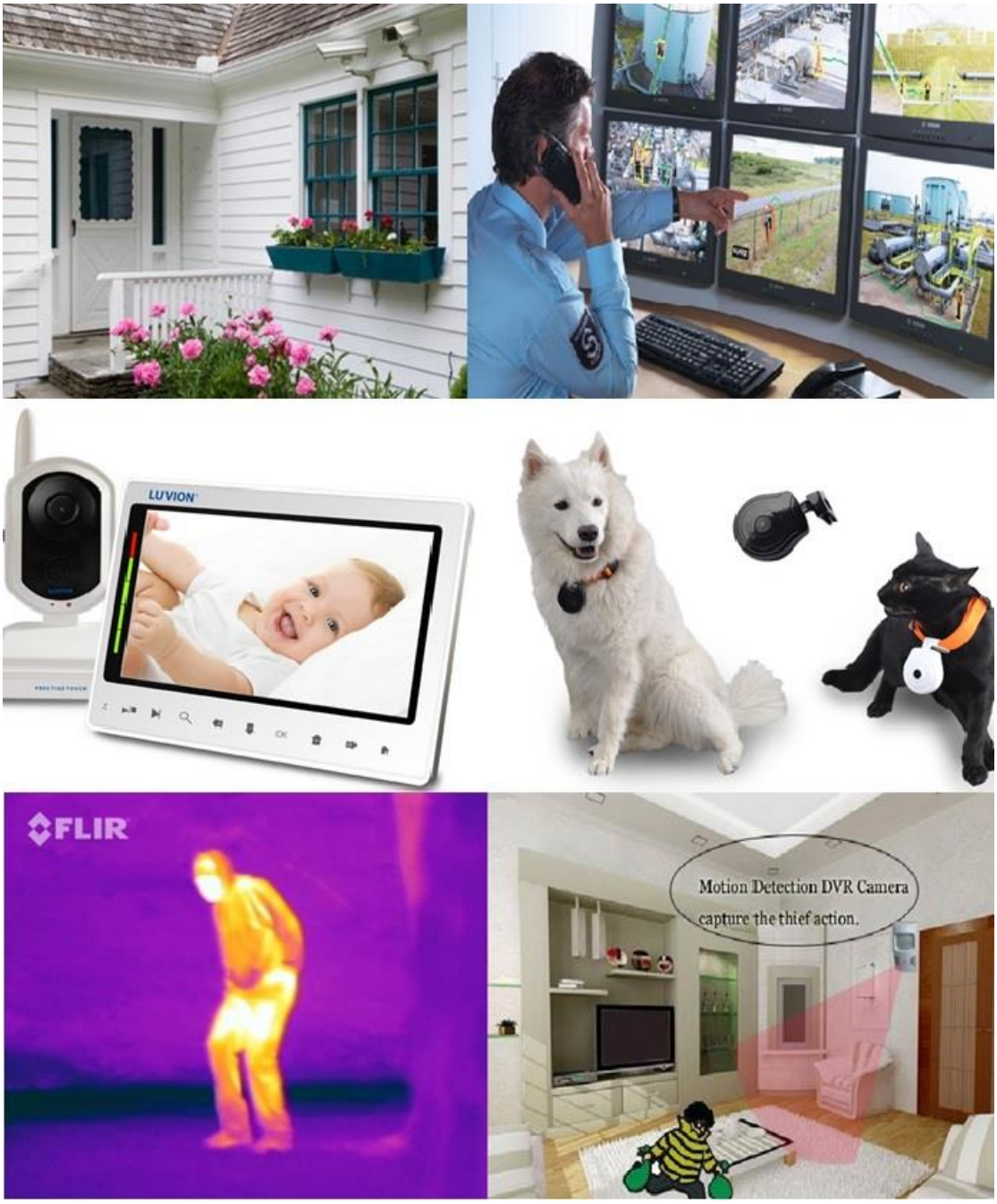

Figure 1 – Camera applications in real life

1.3 Existing types of camera

Figure 2 – Analog camera and how it works

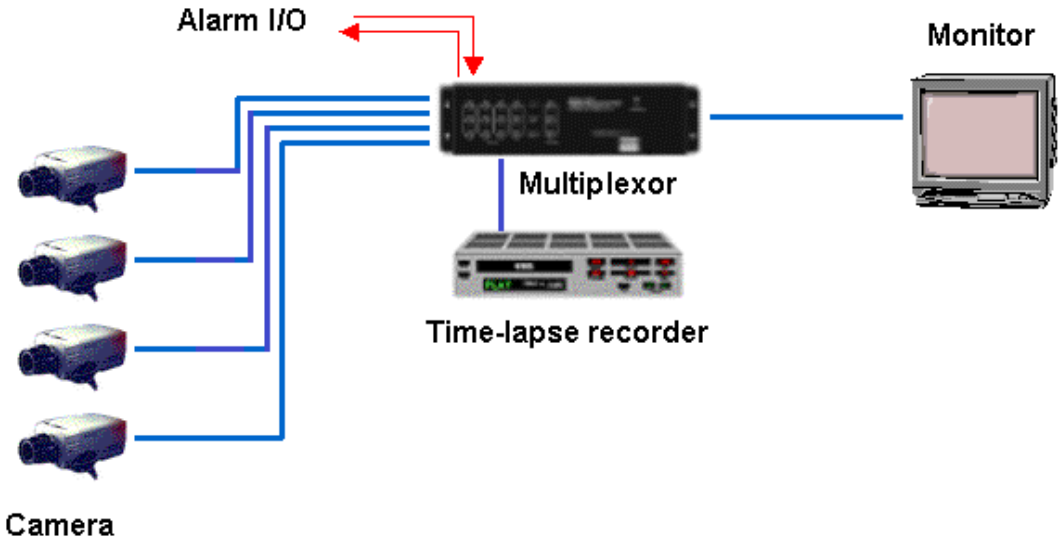

Figure 3 – Analog camera and how it works

There are two popular analog standards in the world for television: NTSC (e.g. as used in North America and Japan) and PAL (used in many other countries, especially in Europe). There is also SECAM but it is not a common standard in CCTV world.

Disadvantage:

- Required to convert the video back to analog in order to transmit.
- Do not have capabilities above the NTSC/PAL standards.
- Only record in the moment activity for future access – cannot

broadcast live information for monitoring purposes.

- Low video quality.
- Relied on human reliability.

1.4 Proposed Solution

Figure 4 – IP camera system

Ip camera is a proposed solution which provides a higher standard camera system. It is undeniable that it brings us many benefits from switching from analog cameras to Ip cameras:

Firstly, the cost for installing an Ip camera system are now more affordable

- More reliable than the old analog system.
- Easier installing or adding cameras.
- Higher resolution images and videos.
- Encryption & authentication: IP cameras offer secure data transmission through encryption and authentication methods such as WPA, WPA2, TKIP, and AES.
- Flexibility: IP cameras can be moved around anywhere on a network (wireless).
- Remote accessibility: live video from selected cameras can be viewed from any computer, anywhere, and also from many mobile smartphones and other devices.

Disadvantages:

- High network bandwidth requirements - it needs a larger bandwidth for sending higher resolution images and videos.
- Higher initial cost per camera.
- High risk of being hacked or controlled by hackers if the video is transmitted over the public internet instead of private LAN network.

1.5 Feature Functions

The system is intended to build contains 3 parts:

- **Module camera Mt9d111 connected with CC3200 Launchpad – Surveillance Camera (SC)**
- **SmartConfig Android App (SAA)**
- **Surveillance Camera Monitor (SCM).**

The camera can connects to a LAN network or the Internet through Wi-Fi connection.

The video - which are sent by the camera through wireless connection, must work probably without errors.

Using the window application monitor, user can watch what is happening at place where camera was set up.

The window application monitor provide user an option to view saved videos in the using computer.

1.6 Project Team

No.	Name	Role	Contact
1	Trương Vĩnh Lân	Supervisor	**Email**: lantv@fpt.edu.vn
2	Nguyễn Tư Duy	Leader	**Email**: duyntse60775@fpt.edu.vn
3	Hoàng Minh Thắng	Member	**Email**: thanghmse60754@fpt.edu.vn
4	Đỗ Văn Thanh Tú	Member	**Email:** tudvt60500@fpt.edu.vn
5	Vũ Quế Anh	Member	**Email**: anhvq00840@fpt.edu.vn

Table 1 – Project team

2. SYSTEM PROJECT MANAGEMENT PLAN (SPMP)

2.1 Problem Definition

2.1.1 Name and Main Points of Capstone Project

English: Surveillance Camera.

Vietnamese: Camera giám sát.

The system requires design a camera comes along with an application monitor in computer and an Android App for camera network configuration. The design for camera uses CC3200 Launchpad from Texas Instruments, camera module mt9d111, program on hardware.

2.1.2 Problem Abstract

It is common knowledge that IP cameras cost more than analog cameras. To make people switch, IP needs to be either significantly cheaper or offer significant economic benefits that analog cannot offer. Also, components for building an IP camera is not cheap at all, not to mention the fact that the video must be transmitted over the network connection without dropping frames or any errors.

2.1.3 Project Overview

2.1.3.1 The current System

The most common video camera that are using all around the world is analog, it has a majority market share in the past few years. Simply defined, an analog camera captures information (videos and images) and sends the video signal in a format that can be viewed on an analog television and/or recorded over storage disks, DVRs. That is where it ends. Although it is widely used, it also has some drawbacks:

- Low resolution.
- Old technology.
- Very sensitive to interference and ground loops.
- Long distance applications are more difficult; if your surveillance needs encompass a wide area, analog cameras may not be your best choice. Analog cameras generally do not accommodate big

distances, and getting them to work over broad ranges can be difficult. Wireless analog is very unreliable and poor quality, and underground cables are extremely prone to lightning and a whole host of other problems.

2.1.3.2 The proposed System

There is the need of invention of a new system of camera that is more reliable and easier to set up as well as have many advantages:

- More secure.
- Uses less devices compared to the analog camera system.
- Real time streaming and recording.
- Flexible for setting up in different conditions.

We design the system with good hardware and platform. The camera has a compact design and high mobility.

2.1.3.3 Boundaries of the System

The system which is intended to build contains **3 parts:**

- **Surveillance Camera (SC)**: The camera(s) uses for electronic motion picture acquisition and transmit video through network connection.
- **Surveillance Camera Monitor (SCM):** A windows application for users to monitor the cameras through network connection.
- **SmartConfig Android App (SAA):** An Android application for users to connect cameras to the network.

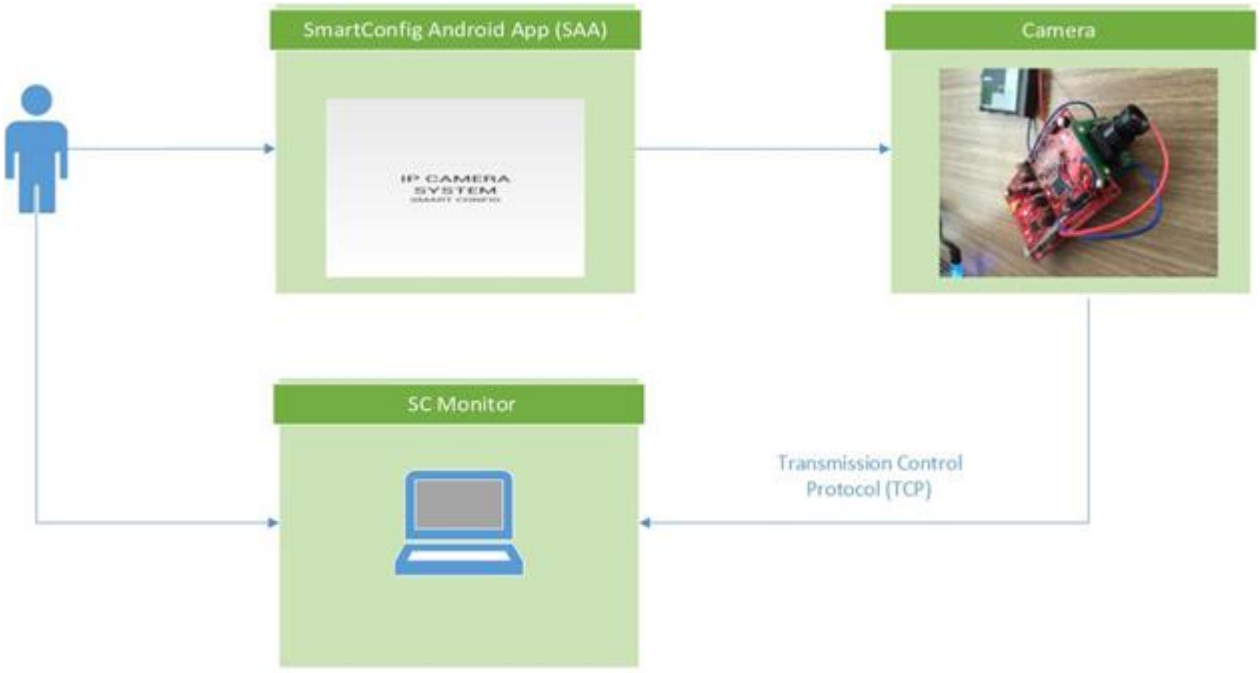

Figure 5 – Boundaries of the System

2.2 Overview

2.2.1 System Process Model

Project is developed under Iterative and Incremental development model.

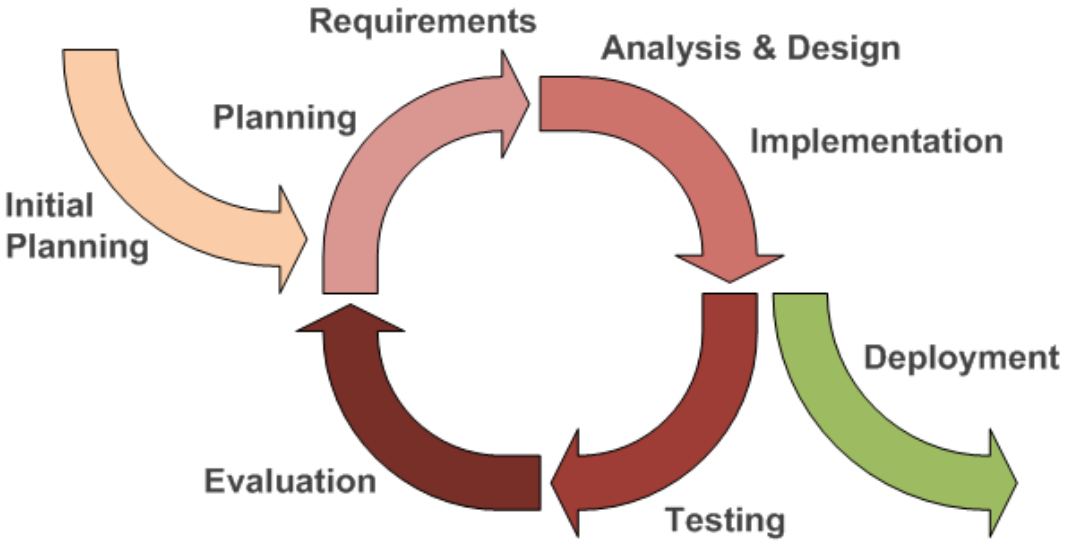

Figure 6 – Iterative and Incremental development model

Iterative and incremental software development is a method of software development that is modeled around a gradual increase in feature additions and a cyclical release and upgrade pattern.

Iterative and incremental software development begins with planning and continues through iterative development cycles involving continuous user feedback and the incremental addition of features concluding with the deployment of completed software at the end of each cycle.

Advantages when applying this model in our project:

- Some working functionality can be developed quickly and early in the life cycle.
- Results are obtained early and periodically.
- Parallel development can be planned.
- Progress can be measured.
- Less costly to change the scope/requirements.
- Testing and debugging during smaller iteration is easy.
- Risks are identified and resolved during iteration; and each iteration is an easily managed milestone.
- Easier to manage risk - High risk part is done first.
- With every increment operational product is delivered.
- Issues, challenges & risks identified from each increment can be utilized/applied to the next increment.
- Risk analysis is better.
- It supports changing requirements.
- Initial Operating time is less.
- Better suited for large and mission-critical projects.
- During life cycle software is produced early which facilitates customer evaluation and feedback.

2.2.2 Roles and Responsibilities

No.	Name	Role	Responsibilities
1	Phan Duy Hùng	Ext Supervisor	- Support technical issue - Specify requirement - Control the development process
2	Nguyễn Tư Duy	Team Leader, Developer, Tester	- Managing process - Assign tasks - Designing hardware - GUI design - Create test cases - Coding - Clarifying requirements - Support technique - Prepare documents
3	Hoàng Minh Thắng	Team Member, Developer, Tester	- GUI Design - Prepare hardware - Coding - Testing - Create test cases - Prepare documents
4	Đỗ Văn Thanh Tú	Team Member, Developer, Tester	- GUI Design - Testing - Prepare documents

5	Vũ Quế Anh	Team Member, Tester	- Testing - Prepare documents

Table 2 – Roles and Responsibilities

2.2.3 Tools and Techniques

2.2.3.1 Camera development

Hardware

The camera hardware can be classified into the following parts:

- **Camera module:** MT9D111
- **Mainboard:** Ti CC3200 Launchpad
- **Wireless Communication** : Integrated Wifi module
- **Addition hardware**:
  - Program/Debug Devices

Software

- **Operating System:** Windows 7 Professional
- **Flash tool** : CCS Uniflash
- **Terminal Emulator :** TeraTerm 4.84
- **IDE :** Code Composer Studio 6.0.1

2.2.3.2 Android application development

- **IDE**: Eclipse with Android Developer Tool add on
- **Technologies**: JAVA

2.2.3.3 Windows application development

- **Wireless Communication** : Network connection
- **Technologies** : .NET Framework 4.0
- **IDE** : Microsoft Visual Studio 2013 Community

2.2.3.4 Communication

By Facebook, Gmail, Skype and in class with Supervisor.

2.3 Project Management Plan

2.3.1 Iteration

Phase /Iteration	**Description**	**Deliverables**	**Resource needed**	**Dependencies and Constraints**	**Risks**
Preliminary study and planning	- Study how to program on CC3200 Launchpad - Study how to work with camera module MT9D111. - Identify and research algorithm and devices - Make project's plan	- Notes about CC3200 Launchpad - Project's plan	40 man-days	N/A	- Lack of information about hardware.
Prepare hardware	- Find materials and components supplier.	- Component devices	15 man-days	N/A	- The supplier not supply the material on time. - Materials can be broken due to inappropriate uses. - Lack of money. - Problems with shipping.

Develop camera firmware	- Implement camera firmware Implement an Android App for users to connect cameras to specific Wi-Fi network	- Completed camera firmware code - Anroid App for smartconfig	75 man-days	Depend on **"Preliminary study and planning"** and **"Prepare hardware"**	- Components can be broken due to wrong settings code - Problems with components may occur due to the producer's errors.
Develop window application	- Implement an window application for users to view and control cameras via network connection	- Window application	20 man-days	- N/A	- Lack of experience in designing and developing a software for controlling cameras.
Integration and system test for the behavior in system	- Create test plan, test cases and expected result - Execute test cases and collect test result	- Integrated system - Test plan, test cases and test result	20 man-days	Depend on **"Develop window application"** and **"Develop camera"**	- Lack of experience in creating test plan and test cases.

Table 3 – Project Iterations

2.3.2 Iteration detail

2.3.2.1 Phase 1: Preliminary study and planning

Task	Description	Author
- Study about CC3200 Launchpad	- Study about CC3200 Launchpad	DuyNT, AnhVQ
- Study about camera module MT9D111	- Study about camera module MT9D111	ThangHM, TuDVT
- Identify and research about Ip camera	- Choose which end devices should be made - Research about needed design	DuyNT, ThangHM
- Planning	- Make plan, schedule and assign task - Identify risk and how to manage	DuyNT
- Document	- Make team's report: Introduction, SPMP	AnhVQ, TuDVT

Table 4 – Phase 1- Preliminary study and planning

2.3.2.2 Phase 2: Prepare hardware

Task	Description	Author
Identify requirement and planning	- Which devices should be made - What components are needed	DuyNT, ThangHM, AnhVQ, TuDVT
Buy material	- CC3200 Launchpad - Camera module MT9D111 - Other materials	ThangHM, DuyNT
Document	- Make team's report: SRS, SDD	ThangHM, DuyNT, AnhVQ, TuDVT

Table 5 – Phase 2 - Design, prepare hardware

2.3.2.3 Phase 3: Develop camera firmware

Task	Description	Author
Implement algorithms in camera firmware	- Implement algorithms in camera firmware	DuyNT, ThangHM,
Implement Android App for smartconfig	Implement Android App for smartconfig	DuyNT
Testing	- Test if all devices work correctly	AnhVQ, DuyNT
Document	- Update SRS, SDD	ThangHM, DuyNT, AnhVQ, TuDVT

Table 6 – Phase 3 - Develop camera firmware

2.3.2.4 Phase 4: Develop window application

Task	Description	Author
- Identify requirement	- Which feature/services the application should provide	DuyNT, ThangHM, TuDVT
- Implement communication	- Implement how application communicates with cameras.	DuyNT, ThangHM
- Implement GUI and Interaction	- Implement application on windows to help users control cameras.	ThangHM, TuDVT
-Testing	- Test system behavior and performance	AnhVQ, TuDVT
- Document	- Update SRS, SDD;	ThangHM, DuyNT, AnhVQ, TuDVT

Table 7 – Phase 4 - Develop window application

2.3.2.5 Phase 5: Integration and system test for the behavior in system

Task	Description	Author
- Identify requirement and planning	- Which feature the system can be improved	DuyNT, ThangHM, AnhVQ, TuDVT
- Test performance	- Check the performance of system	DuyNT, ThangHM,
- Create test case	- Check the behavior of system when risk occurred	DuyNT, ThangHM,
- Execute test case	- Create test cases to cover	DuyNT, AnhVQ,
- Check the result and fix bug	- Collect test result - If any bug occurred, fixed them	DuyNT, ThangHM, AnhVQ, TuDVT
- Document	- Add test plan and test result - Create User Manual	AnhVQ, TuDVT

Table 8 – Phase 5 - Integration and system test for the behavior in System

2.3.3 Task Sheet: Assignments and Timetable

See the Task sheet hard-copy.

2.4 Convention Rules

The following rules follow:

- The standard rules for developing application using C/C++ (http://users.ece.cmu.edu/~eno/coding/CCodingStandard.html)
- MSDN's C# Coding Conventions (C# Programming Guide) (http://msdn.microsoft.com/en-us/library/vstudio/ff926074%28v=vs.110%29.aspx)

3. SYSTEM REQUIREMENTS SPECIFICATIONS (SRS)

3.1 User Requirement Specification

There are 5 main functions:

- **Connect cameras to network:** User can use SmartConfig Android App to connect cameras to network.
- **Watch videos:** User can watch videos which are transmitted by connected cameras.
- **Auto save videos**: The video which are transmitted by connected cameras will be saved to the computer automatically.
- **Control cameras**: User can change camera's setting such as name, port number and description. User can also add or remove a camera.
- **View recorded videos:** User can watch the recorded videos which are saved automatically when cameras stream to the computer. User can choose the time and the camera they want to watch. The video control is also needed for improving user's convenience, which allow user to play the video forward with 2X and 4X speed, it is the same for the backward playing.

3.2 System Requirement Specification

3.2.1 External Interface Requirements

3.2.1.1 User Interface

The interface must be designed to be satisfied the following requirement:

- Be simple and user-friendly.
- Meet all the main functions and easily to identify each of functions.
- Use obvious icons to avoiding misunderstanding.

3.2.1.2 Hardware Interface

The hardware interface the camera using must be designed to be satisfied

the following requirement:

- Low-cost hardware module.
- Easy to consume.
- Easy to replace for maintenance.

3.2.1.3 Communication Protocols

The cameras and the windows application must use network connection to connect each other, using integrated Wi-Fi module in CC3200 Launchpad. The video is sent to the computer using TCP protocol.

3.2.2 Functional Requirements

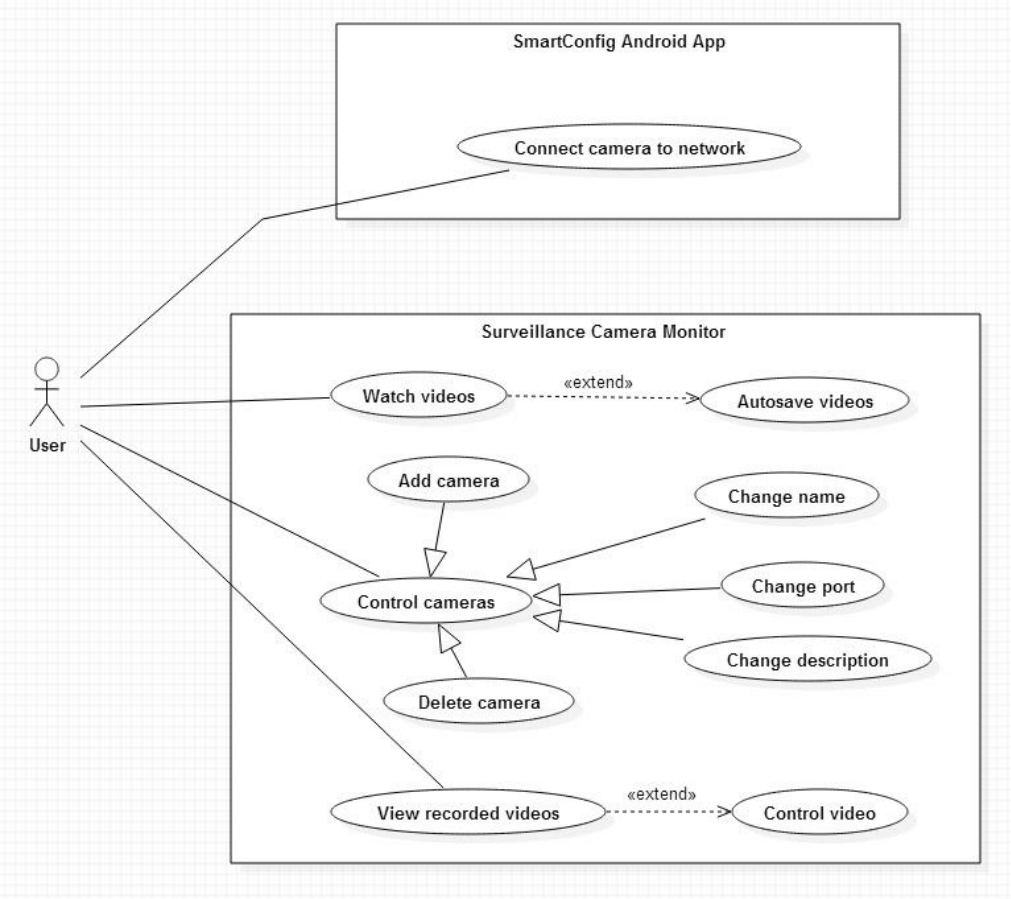

Figure 7 – Use case Diagram

3.2.2.1 Connect cameras to network

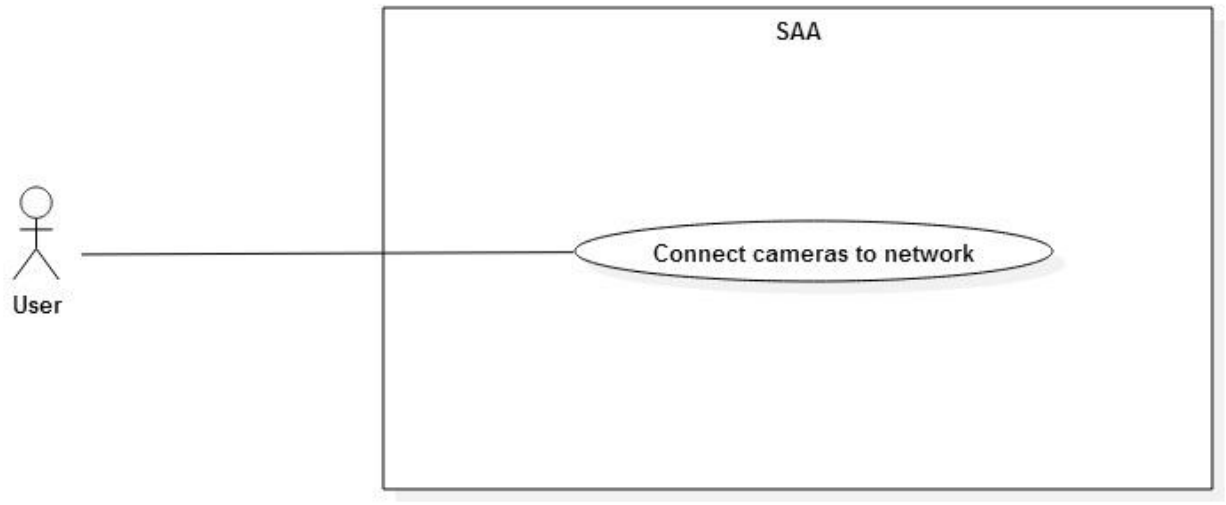

Figure 8 –Connect cameras to network use case Diagram

USE CASE - UC001 – Connect Cameras To Network			
Use-case No.	UC001	**Use-case Version**	1.0
Use-case Name	Connect cameras to network		
Author	Nguyễn Tư Duy		
Date	2015/02/11	**Priority**	Normal

Actor:
User
Summary:
- This use case allow user to use SmartConfig Android App (SAA) to connect camera(s) to a specific network.

Goal:
- Connect camera to the network.

Triggers:
- User click on "Start" button on the application.

Preconditions:
- User's Android devices connected to a network.

Post Conditions:
- Camera is on the device list of the Android App.
- Camera connected to the network.

Main Success Scenario:

Step	Actor Action	System Response
1	Open the App	App is opened. Welcome screen is shown.
2	Click on "SmartConfig" tab.	Change to "SmartConfig" tab. Name of the network is auto filled with the name of the network which Android device is currently connected to.
3	Click on "Start" button.	Start searching for cameras match the entered name. Send command to those cameras to connect to the network A message box appears with content: "New device found! You can view it at the "Devices" tab"
4	Click on "OK" button.	Return to "SmartConfig" tab.
5	Click on "Devices" tab.	Change to "Devices" tab. List of the connected cameras is shown.

Alternative Scenario:

Step	Actor Action	System Response
3a	Click on "Cancel" button during the searching	Return to "SmartConfig" tab. No change is made.

Exceptions:

Step	Actor Action	System Response
1	Enter wrong password.	A message box appears with message "You did not enter a password. Whould you like to continue?"

2	No cameras are found.	A message box appears with message "No new device has been found"
3	No network connection.	A message box appears with message "No Wifi connection detected. What would you like to do?"

Relationship:
N/A

Table 9 – Connect cameras to network use case Description

3.2.2.2 Watch Videos Function

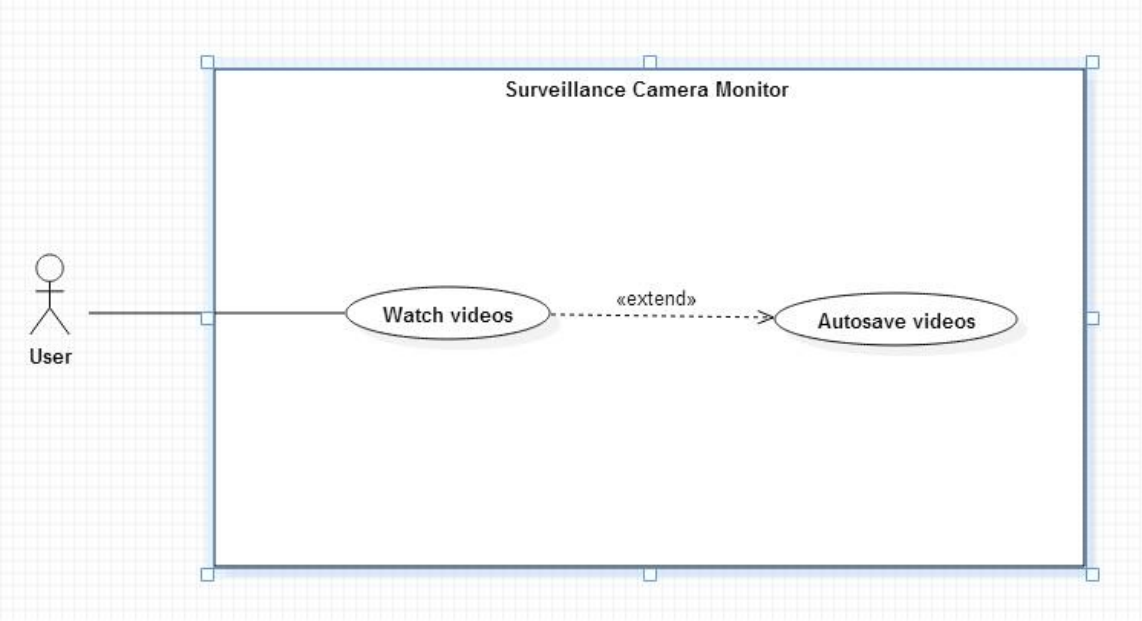

Figure 9 – Watch Videos Use case Diagram

USE CASE - UC002 – Watch Videos			
Use-case No.	UC002	**Use-case Version**	1.0
Use-case Name	Watch Videos		
Author	Nguyễn Tư Duy		
Date	2015/02/11	**Priority**	Normal

Actor:

User

Summary:

- This use case allow user watch videos from cameras after setting appropriate camera information and connecting to the camera through network connection.

Goal:

- The windows application successfully connect to cameras via network connection.

Triggers:

- User click on “Start” below the video box.

Preconditions:

- Camera is ON.
- Computer connected to a network.
- Camera is connected to the same network with the computer.

Post Conditions:

- The video which transmitted from the added camera is received and played.

Main Success Scenario:

Step	Actor Action	System Response
1	User open the application	Application is opened. Camera’s area of application is blank if the application is opened for the first time or there was no camera connection in the last time application was opened.
2	User click on “Setting” button	Setting screen is opened Camera list is shown (include camera’s information such as name, port number

		and description)
3	User click on "Add" button	Adding form appears, which ask user for camera's information: name, port number and description (not require).
4	User click on "OK" button	Application change back to Setting screen. New camera appears on the camera list.
5	User close Setting screen	Setting screen is closed. A camera box appear in camera's area of the application.
6	User click on "Start" button	The computer starts receiving video from the added port. Video in the video box starts playing. The video is automatically saved in the local disk in order to watch again later.

Alternative Scenario:

N/A

Exceptions:

No	Actor Action	System Response
1	Port number is left blank.	A message box appears with message *"Invalid port number! Please check then try again!"* in the top of the form.
2	The port number is already used.	A message box appears with message "*The port is already used for another camera, please enter another port.*".
3	The name of the camera is left blank.	Camera is added with the default name: CameraX, X is the number of the camera.

Relationship: Autosave videos.

Table 10 – Watch Videos Use case Description

3.2.2.3 Control Cameras

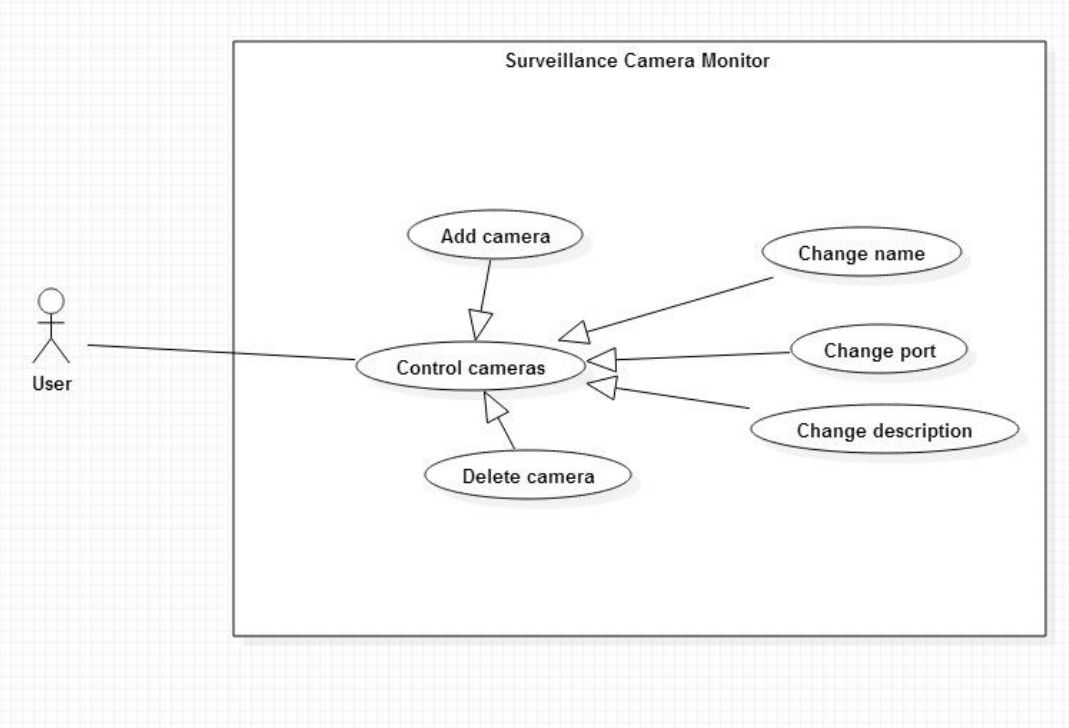

Figure 10 – Control Cameras Use case Diagram

Add Camera

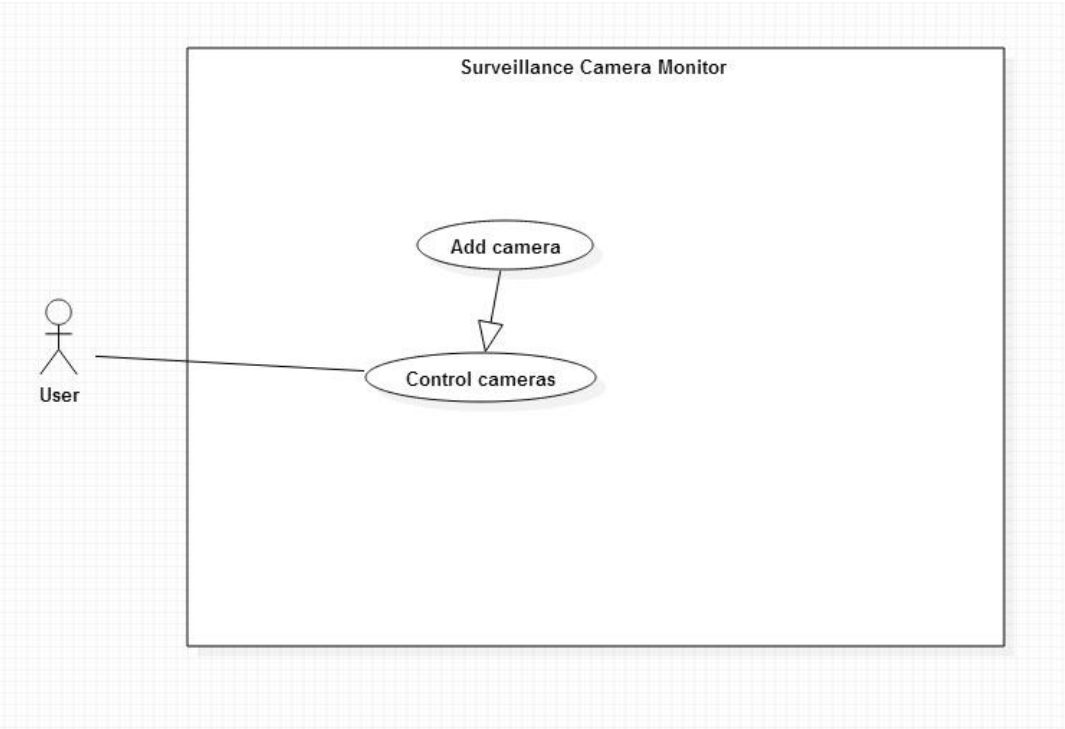

Figure 11 – Add Camera Use case Diagram

USE CASE - UC003 – Add Camera			
Use-case No.	UC003	**Use-case Version**	1.0
Use-case Name	Add Camera		
Author	Hoàng Minh Thắng		
Date	2015/02/15	**Priority**	Normal

Actor:

User

Summary:

- This use case allow user adding cameras.

Goal:

- A new camera is added to the list of cameras.

Triggers:

- User click on "Add" button on Setting screen, which is opened by clicking on "Setting" button.

Preconditions:

- Camera is ON.
- Computer connected to a network.
- Camera is connected to the same network with the computer.

Post Conditions:

- A new camera is added to the camera list.
- A camera box is added for the new camera.

Main Success Scenario:

Step	Actor Action	System Response
1	User open the application	Application is opened. Camera's area of application is blank if the application is opened for the first time or there was no camera connection in the last time application was opened.
2	User click on "Setting" button	Setting screen is opened Camera list is shown (include camera's information such as name, port number and description)

3	User click on "Add" button	Adding form appears, which ask user for camera's information: name, port number and description (not require).
4	User click on "OK" button	Application change back to Setting screen. New camera appears on the camera list.
5	User close Setting screen	Setting screen is closed. A camera box appear in camera's area of the application.

Alternative Scenario:

No	Actor Action	System Response
4a	Click "Cancel" or "X" button at the top right corner of the Setting screen.	Setting screen is closed. No camera is added. No new camera box is created.

Exceptions:

No	Actor Action	System Response
1	Port number textbox is left blank.	Appear message *"Invalid port number! Please check then try again!"* in the top of the form.
2	Entered port number is already used.	A message box appear appears with content: "This port is already used, try enter another port!"
3	The name of the camera is left blank.	Camera is added with the default name: "CameraX", X is the number of the camera.
4	The entered name is already exist.	A message box appear appears with content: "This name is already used, try enter another name!"

Relationship: Control cameras.

Table 11 – Add Camera Use Case Description

Change Name

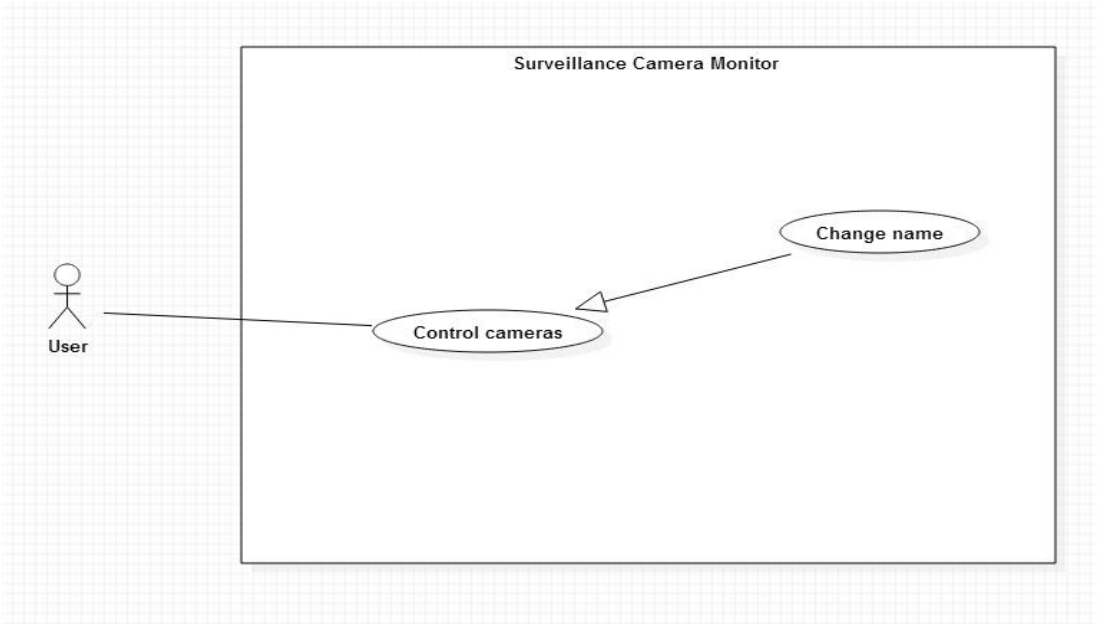

Figure 42 – Change Name Use Case Diagram

USE CASE - UC004 – Change Name			
Use-case No.	UC004	**Use-case Version**	1.0
Use-case Name	Change Name		
Author	Hoàng Minh Thắng		
Date	2015/02/15	**Priority**	Normal

Actor:

User

Summary:

- This use case allow user to change the current name of an added camera.

Goal:

- Name of the chosen camera is changed.

Triggers:

- User click on "Edit" button on Setting screen, which is opened by clicking on "Setting" button.

Preconditions:

- Camera(s) is/are added and shown in the list of camera.

Post Conditions:

- Name of the camera is changed.

Main Success Scenario:

Step	Actor Action	System Response
1	User open the application	Application is opened. Camera's area of application is blank if the application is opened for the first time or there was no camera connection in the last time application was opened.
2	User click on "Setting" button	Setting screen is opened. Camera list is shown (include camera's information such as name, port number and description).
3	Choose a camera on the list. Click "Edit" button.	Camera's information form is opened. Information of the chosen camera is shown in textboxes.
4	Change the name of the camera.	A message box appear appears with

	Click "Save" button.	content: "Update successful!"
5	Click "OK" button.	The name of the camera is changed in the list.
6	Click "Cancel" or "X" button at the top right corner of the Setting screen.	The name of the camera in the camera box is changed.

Alternative Scenario:

Step	Actor Action	System Response
3a	Click on "Edit" button.	A message box appear appears with content: "Choose a camera on the list!"
4a	- Click "Cancel" or "X" button at the top right corner of the camera's information form	The form is closed, no change is executed.

Exceptions:

No	Actor Action	System Response
1	The name of the camera is left blank.	Camera is added with the default name: "CameraX", X is the number of the camera.
2	The entered name is already exist.	A message box appear appears with content: "This name is already used, try enter another name!"

Relationship:

Control cameras.

Business Rules:

N/A

Table 12 – Change Name Use Case Description

Change Port

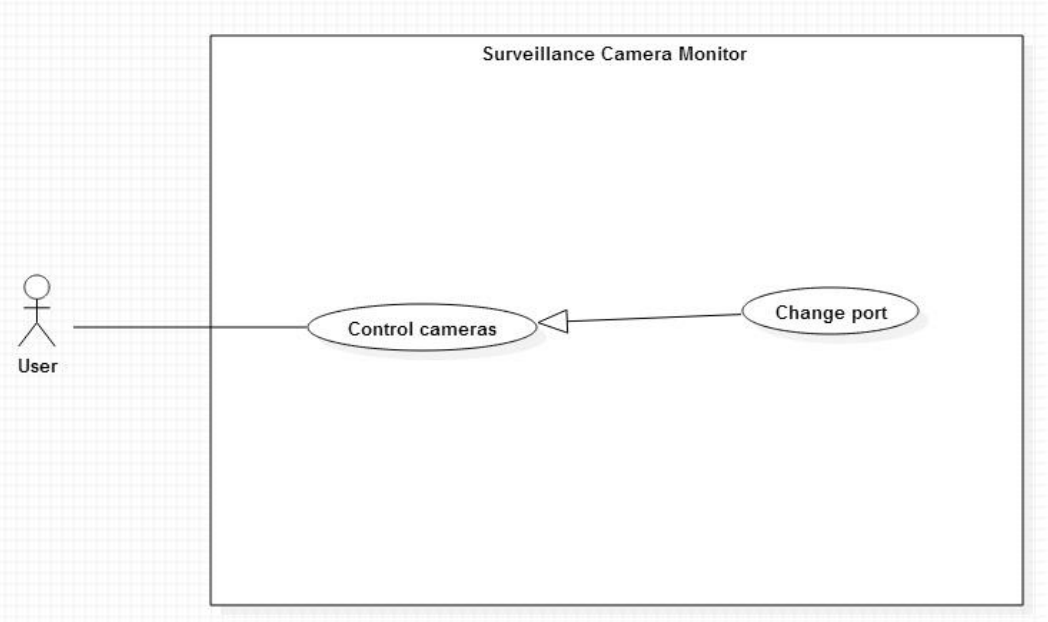

Figure 13 – Change Port Use Case Diagram

USE CASE - UC005 – Change Port			
Use-case No.	UC005	**Use-case Version**	1.0
Use-case Name	Change Port		
Author	Hoàng Minh Thắng		
Date	2015/02/15	**Priority**	Normal

Actor:

User

Summary:

- This use case allow user to change the current port of an added camera.

Goal:

- Port of the chosen camera is changed.

Triggers:

- User click on "Edit" button on Setting screen, which is opened by clicking on "Setting" button.

Preconditions:

- Camera(s) is/are added and shown in the list of camera.

Post Conditions:

- Port of the camera is changed.

Main Success Scenario:

Step	Actor Action	System Response
1	User open the application	Application is opened. Camera's area of application is blank if the application is opened for the first time or there was no camera connection in the last time application was opened.
2	User click on "Setting" button	Setting screen is opened. Camera list is shown (include camera's information such as name, port number and description).
3	Choose a camera on the list.	Camera's information form is opened. Information of the chosen camera is

	Click "Edit" button.	shown in textboxes.
4	Change the port of the camera. Click "Save" button.	A message box appear appears with content: "Update successful!"
5	Click "OK" button.	The port of the camera is changed in the list.

Alternative Scenario:

Step	Actor Action	System Response
3a	Click on "Edit" button.	A message box appear appears with content: "Choose a camera on the list!"
4a	Click "Cancel" or "X" button at the top right corner of the Setting screen.	The form is closed, no change is made.

Exceptions:

No	Actor Action	System Response
1	The port number textbox is left blank.	A message box appears with content: "Please enter the connection port!"
2	The entered port is already exist.	A message box appears with content: "This port is already used, try enter another port!"

Relationship:

Control cameras.

Business Rules:

- The input port number must be a number in range of 1-65535.

Table 13 – Change Port Use Case Description

Change Description

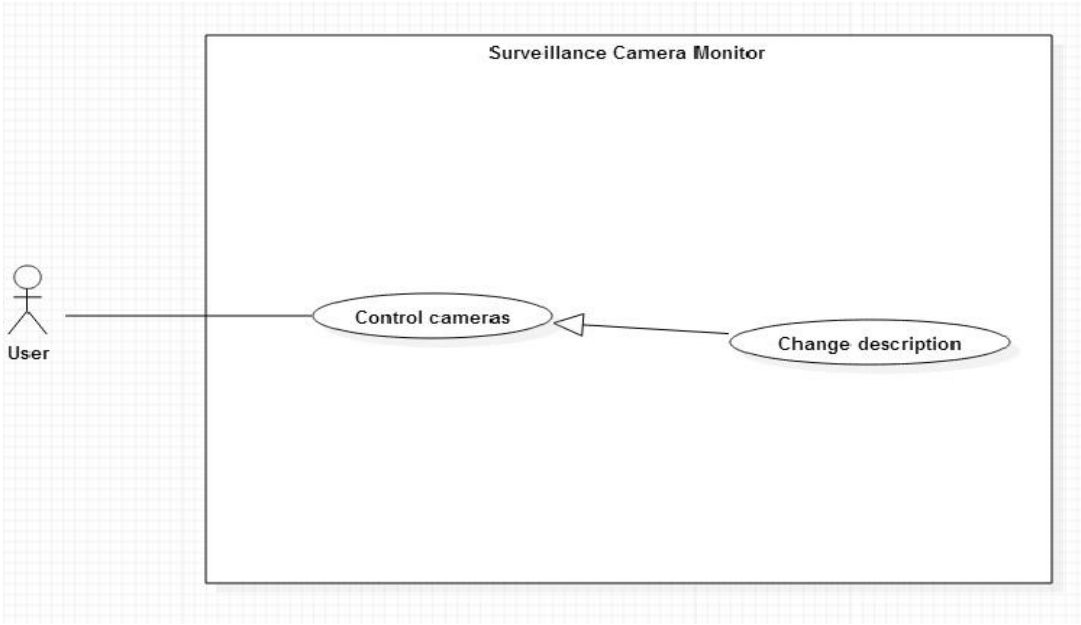

Figure 5 – Change Description Use Case Diagram

USE CASE - UC006 – Change Description			
Use-case No.	UC006	**Use-case Version**	1.0
Use-case Name	Change Description		
Author	Hoàng Minh Thắng		
Date	2015/02/15	**Priority**	Normal

Actor:

User

Summary:

- This use case allow user to change the current description of an added camera.

Goal:

- Description of the chosen camera is changed.

Triggers:

- User click on "Edit" button on Setting screen, which is opened by clicking on "Setting" button.

Preconditions:

- Camera(s) is/are added and shown in the list of camera.

Post Conditions:

- Description of the camera is changed.

Main Success Scenario:

Step	Actor Action	System Response
1	User open the application	Application is opened. Camera's area of application is blank if the application is opened for the first time or there was no camera connection in the last time application was opened.
2	User click on "Setting" button	Setting screen is opened. Camera list is shown (include camera's information such as name, port number and description).

3	Choose a camera on the list. Click "Edit" button.	Camera's information form is opened. Information of the chosen camera is shown in textboxes.
4	Change the description of the camera. Click "Save" button.	A message box appear appears with content: "Update successful!"
5	Click "OK" button.	The description of the camera is changed in the list.

Alternative Scenario:

Step	Actor Action	System Response
3a	Click on "Edit" button.	A message box appear appears with content: "Choose a camera on the list!"
4a	Click "Cancel" or "X" button at the top right corner of the Setting screen.	The form is closed, no change is made.

Exceptions:

N/A

Relationship:

Control cameras.

Business Rules:

N/A

Table 14 – Change Description Use Case Description

Delete Camera

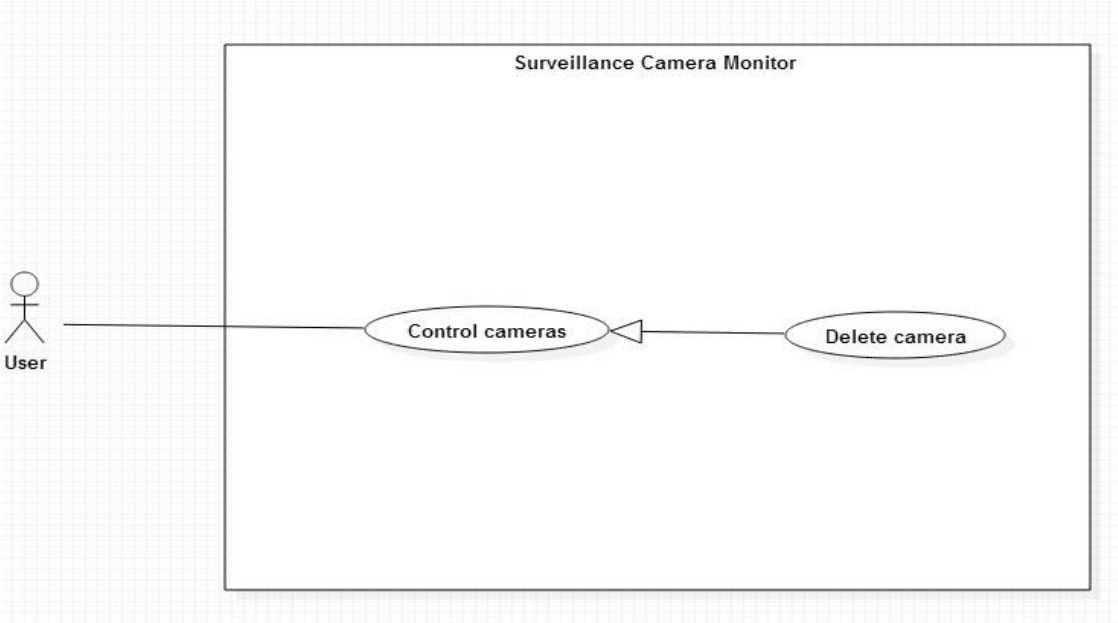

Figure 65 – Delete Camera Use Case Diagram

USE CASE - UC007 – Delete Camera			
Use-case No.	UC007	**Use-case Version**	1.0
Use-case Name	Delete Camera		
Author	Hoàng Minh Thắng		
Date	2015/02/15	**Priority**	Normal

Actor:

User

Summary:

- This use case allow user to delete a camera from the list.

Goal:

- The chosen camera is removed from the list.

Triggers:

- User click on "Delete" button on Setting screen, which is opened by clicking on "Setting" button.

Preconditions:

- The application is opened.

Post Conditions:

- The recorded videos are successful started to play.

Main Success Scenario:

Step	Actor Action	System Response
1	User open the application	Application is opened. Camera's area of application is blank if the application is opened for the first time or there was no camera connection in the last time application was opened.
2	User click on "Setting" button	Setting screen is opened Camera list is shown (include camera's information such as name, port number and description)

3	Choose a camera on the list. User click on "Delete" button	A dialog appears with contents: "Do you want to delete this camera?"
4	User click on "Yes" button	Application change back to Setting screen. The chosen camera is removed from the list.
5	Click "Cancel" or "X" button at the top right corner of the Setting screen.	Setting screen is closed. A camera box appear in camera's area of the application.

Alternative Scenario:

Step	Actor Action	System Response
3a	Click on "Delete" button.	A message box appear appears with content: "Choose a camera on the list!"
4a	Click "No" button.	Return to Setting screen. No change is made.

Exceptions:

N/A

Relationship:

Control cameras.

Business Rules:

N/A

Table 15 – Delete Camera Use Case Description

3.2.2.4 View Recorded Videos Function

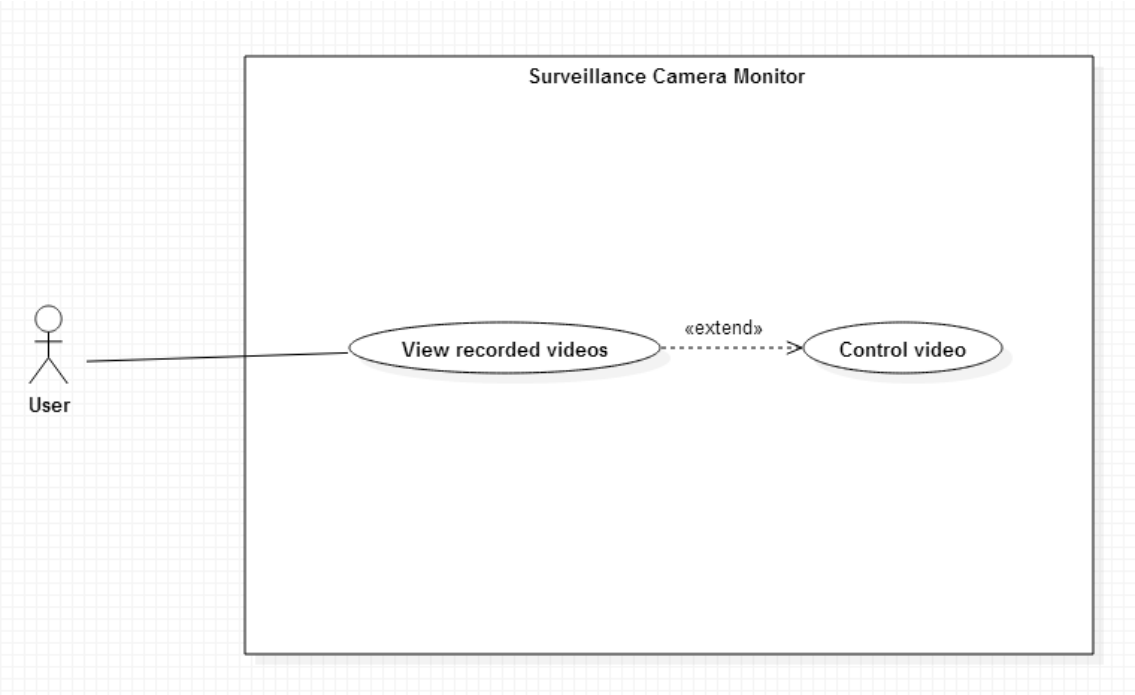

Figure 76 – View Recorded Use Case Diagram

USE CASE - UC008 – View Recorded Videos			
Use-case No.	UC008	**Use-case Version**	1.0
Use-case Name	View Recorded Videos		
Author	Nguyễn Tư Duy		
Date	2015/02/11	**Priority**	Normal

Actor:

User

Summary:

- This use case allow user to play recorded videos.

Goal:

- A record video starts to play.

Triggers:

- User click on “Playback” button on the application.

Preconditions:

- The application is opened.

Post Conditions:

- The recorded videos are successful started to play.
- User can control videos.

Main Success Scenario:

Step	Actor Action	System Response
1	Clicks on “Playback” button.	The playback form appear. “Play”, “<< Play” and “Play >>” button is disable.
2	Click on “date area”. Choose date.	The chosen date is selected and appear on the textbox.
3	Click on “Camera” combo box. Choose a camera.	The chosen camera display on the combo box. Available hour’s button among 24 buttons will be marked with red color in background. The rest of the 24 buttons will be

		disable.
4	Choose an available hour. Click on an available hour button.	The video appears on the left area. "Play" buttons become available.
5	Click on "Play" button.	The video starts playing. "Play >>"and "<< Play" button become available.
6	Click on "Play >>"button.	The video starts playing with 2X speed.
7	Click on "Play >>"button.	The video starts playing with 4X speed.
8	Click on "<< Play" button.	The video starts playing backward.

Alternative Scenario:

Step	Actor Action	System Response
4a	Click on an unavailable hour button.	A message box appear appears with content: "There is no records in this hour!"
6a	Click on "<< Play" button.	The video starts playing with 2X speed.
7a	Click on "<< Play" button.	The video starts playing with 4X speed.

Exceptions:

N/A

Relationship:

View recorded videos, Control video.

Business Rules:

N/A

Table 16 – View Recorded Videos Use Case Description

3.2.3 Non-functional Requirements

3.2.3.1 Reliability

- The cameras have to play 24/7 without errors or disconnecting from network.
- All computers in the network can connect to cameras if the monitor application is installed.

3.2.3.2 Availability

- All computers in the network can connect to cameras if the monitor application is installed.

3.2.3.3 Security

- Sending videos in the network using TCP protocol connection.

3.2.3.4 Maintainability

- Flexible mechanical designing to make it easy to replace any components and modules.
- The CC3200 Launchpad can be ordered easily in TI website, and camera module mt9d111 can be purchased using EBay or Taobao.
- The system is divided into separate modules (SC, SAA, SCM).
- The firmware code and software code is easy to maintain and upgrade.

3.2.3.5 Portability

- N/A

3.2.3.6 Performance

- Module Mt9d111 handling maximum frame rate 15 fps at full resolution (1200x1600), 30 fps in preview mode(800 x 600).
- Cc3200 handling 4 – 5 fps and transmitted image size 640x480 pixel.

4. SOFTWARE DESIGN DESCRIPTION (SDD)

4.1 Design Overview

This document describes technical, board design and user interface design of Smart IP Camera. It includes:

- System architectural design which describes system's model, system's layers, layer's component and the function of each layer.
- The detailed description of components and the relationship between each component.
- Sequence diagram to know how to the system run.
- User Interface design which describes system's screens and how users interact to systems. Each screen includes functions, types of input/ output and event handling.

4.2 System Architecture Design

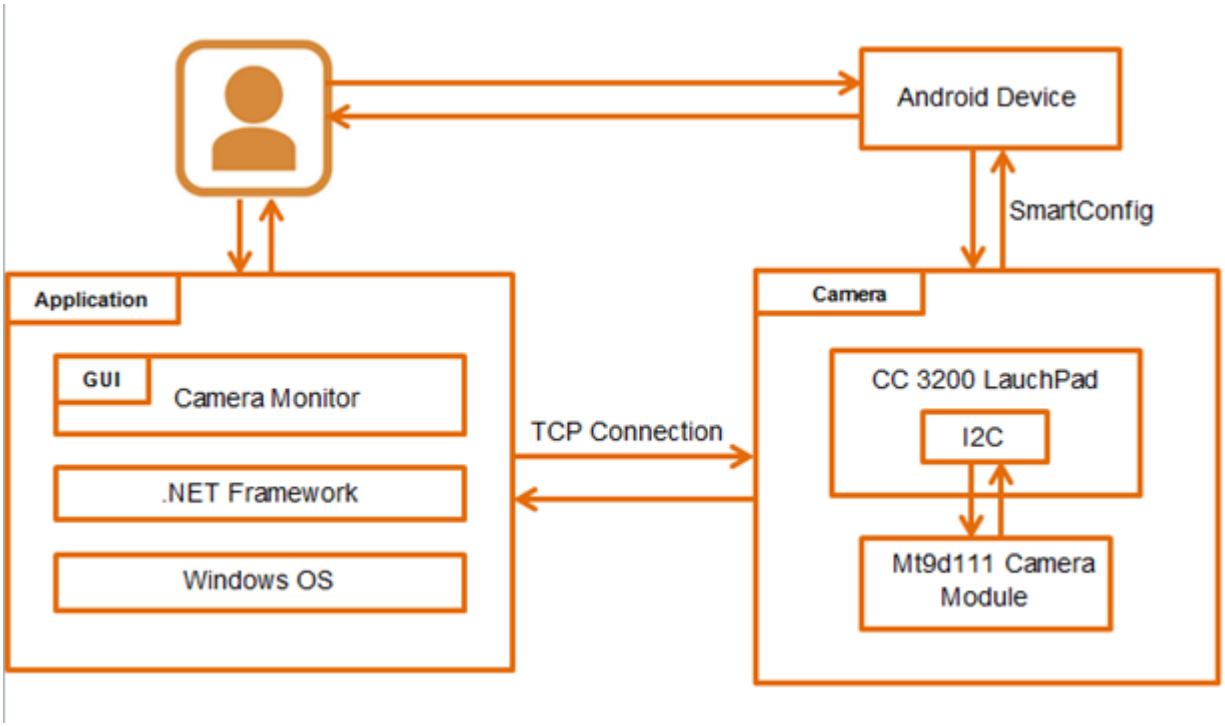

Figure 87 – System Architecture Design

4.3 Component Diagram

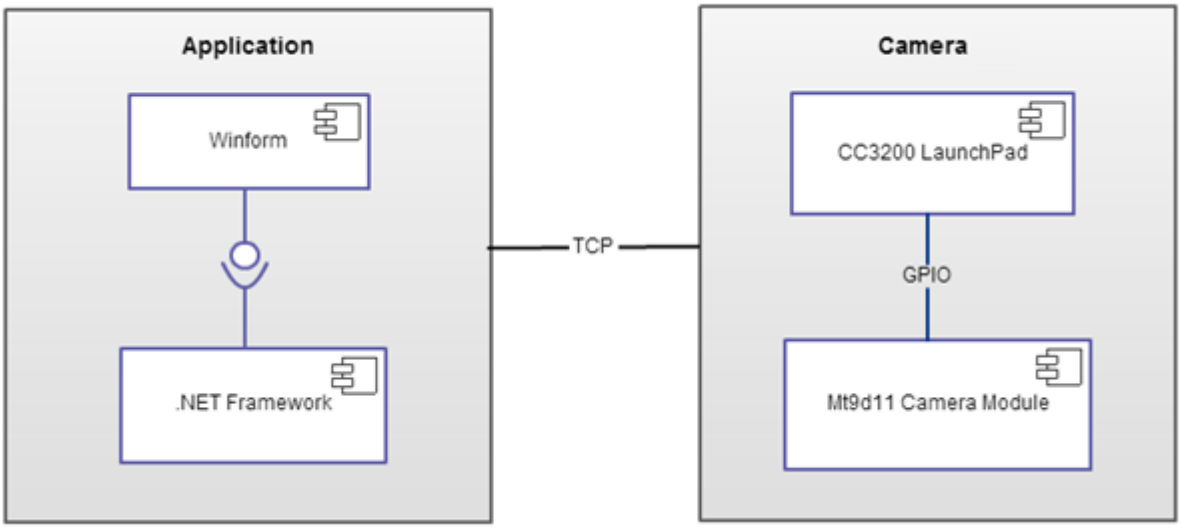

Figure 18 – Component Diagram

4.4 Detailed Design of Components

4.4.1 Hardware

4.4.1.1 CC3200 Launchpad

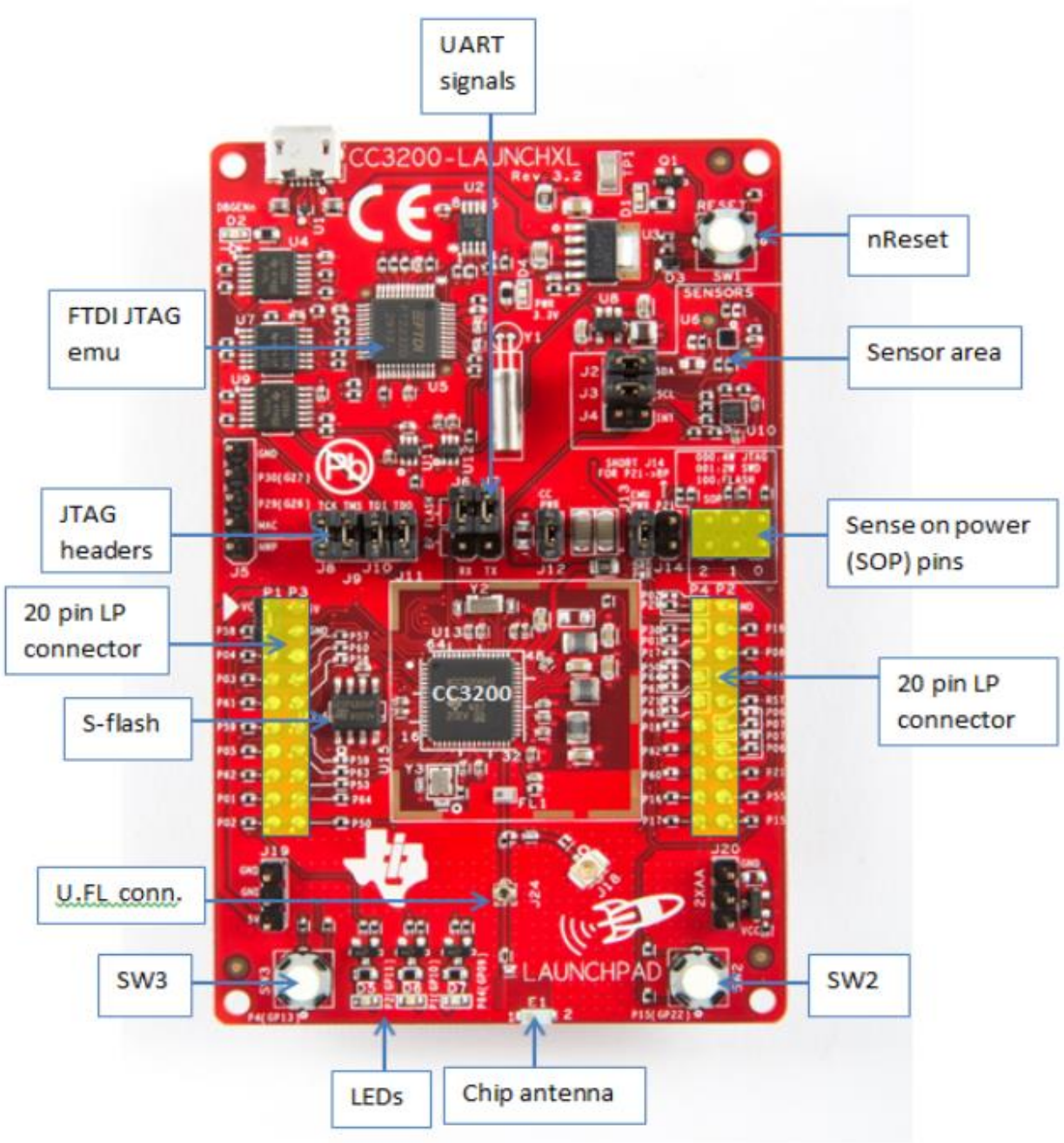

Figure 99 – CC3200 Launchpad Overview

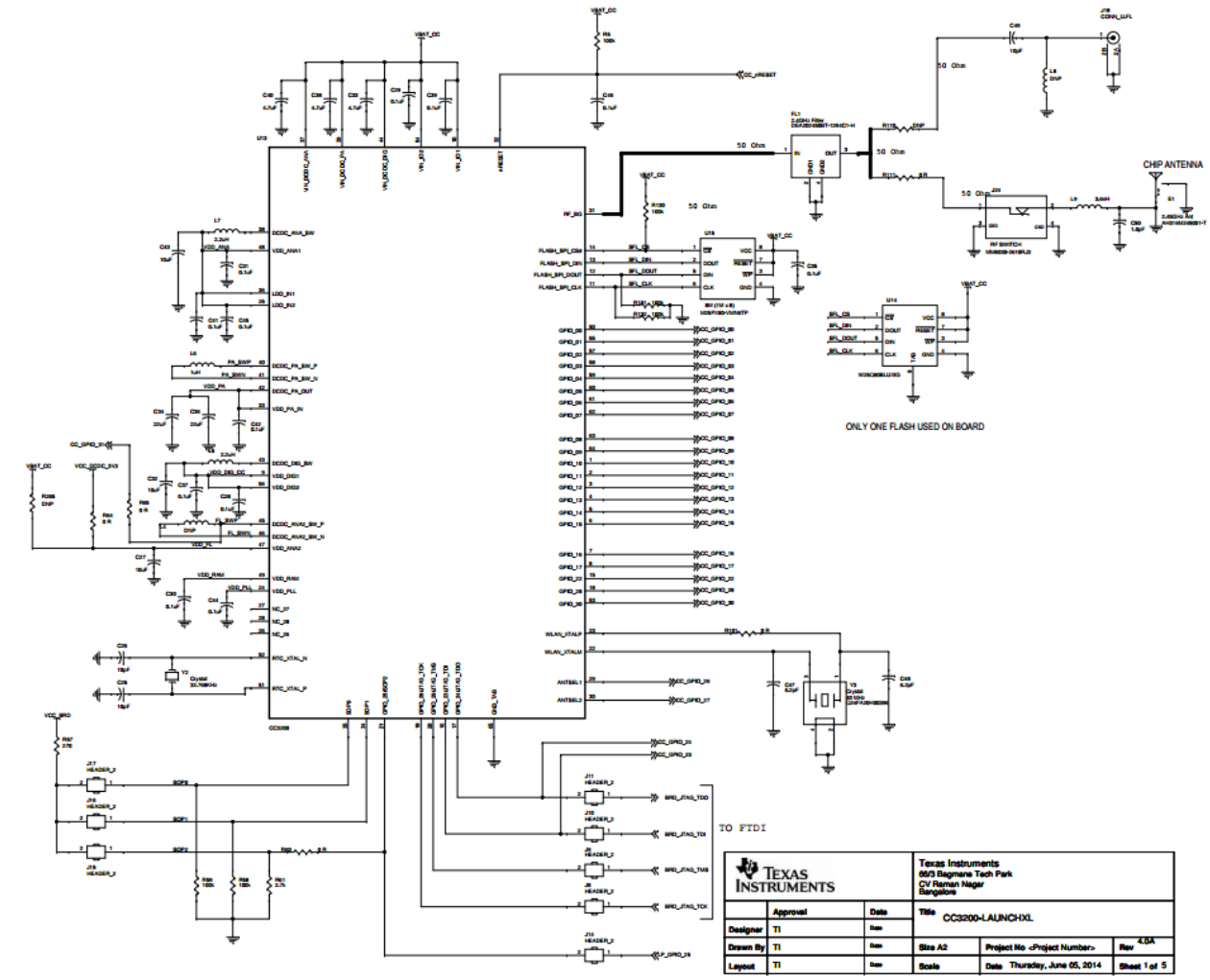

Figure 100 – CC3200 Launchpad Schematic

The high performance CC3200 is the industry's first single-chip Microcontroller (MCU) with built-in Wi-Fi connectivity for the LaunchPad™ ecosystem. Created for the Internet of Things (IoT), the SimpleLink Wi-Fi CC3200 device is a wireless MCU that integrates a high-performance ARM® Cortex®-M4 MCU allowing customers to develop an entire application with a single IC. With on-chip Wi-Fi, internet and robust security protocols, no prior Wi-Fi experience is needed for faster development.

The CC3200 LaunchPad is a low-cost evaluation platform for ARM®

Cortex™-M4F-based microcontrollers. The LaunchPad design highlights the CC3200 Internet-on-a-chip™ solution and WiFi capabilities. The CC3200 LaunchPad also features programmable user buttons, RGB LED for custom applications and onboard emulation for debugging. The stackable headers of the CC3200 LaunchPad XL interface demonstrate how easy it is to expand the functionality of the LaunchPad when interfacing with other peripherals on many existing BoosterPack add-on boards such as graphical displays, audio codec, antenna selection, environmental sensing, and much more.

Features:

- CC3200 Wi-Fi wireless MCU
- USB interface to PC for CCS/IAR using FTDI USB drivers
- Flash update over the USB using SimpleLink Programmer
- 2 20-pin connectors enables compatibility with other TI LaunchPads with added functions
- Standalone development platform featuring sensors, LEDs and push-buttons
- Power from USB for the LaunchPad as well as external BoosterPack
- Operates from 2 AA alkaline batteries
- On-board antenna and U.FL connector selectable using a capacitor re-work
- Supports 4 wire JTAG and 2 Wire SWD
- GNU Debugger (GDB) support over Open On chip debugger (OpenOCD)
- BoosterPack headers

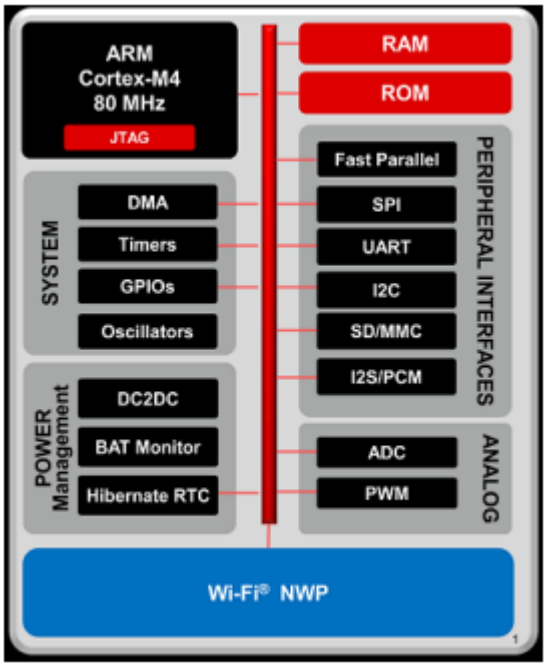

Figure 11 – CC3200 Overview of Peripherals

4.4.1.2 MT9D111 Camera Module

Figure 22 – MT9D111 Camera Module

MT9D111 is a 1/3.2 inch 2-megapixel CMOS image sensor with an integrated advanced camera system. The camera system features a microcontroller (MCU) and a sophisticated image flow processor (IFP) with a real-time JPEG encoder. It also includes a programmable general purpose I/O module (GPIO), which can be used to control external auto focus, optical zoom, or mechanical

shutter.

Features:

- Ultra-low power consumption, progressive scan
- 200 million pixels resolution (1600 H x 1200 V)
- 1/3.2 inch optical format
- frame rate up to 15 fps under full resolution
- Real-time JPEG encoding
- Integrated with processor, with much flexibility
- Many automatic functions including correcting or enhancing images, and fast exposure adjustment
- Integrated 10-bit ADC chip
- Support ITU_R BT.656 (YCbCr), 565RGB, 555RGB, 444RGB and original output data format
- Support JPEG 4:2:2 and 4:2:0 output

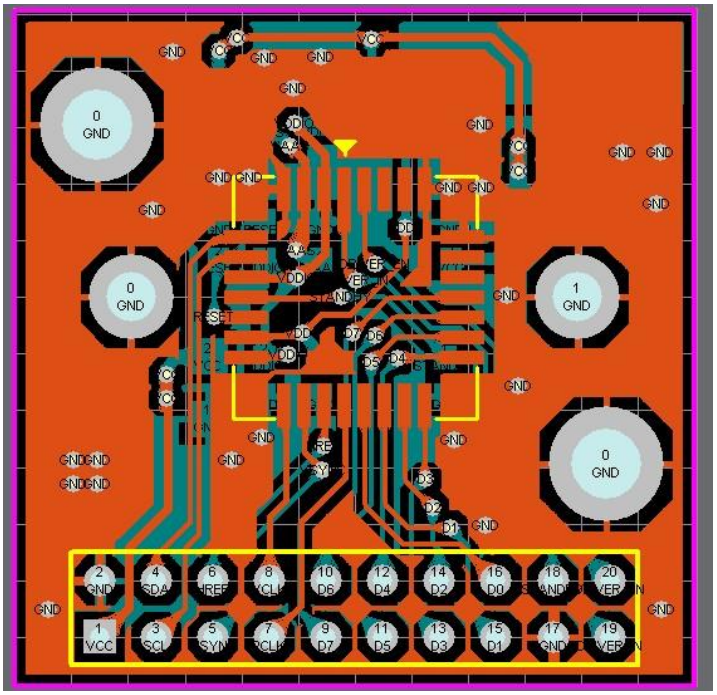

Figure 23 – MT9D111 PIN

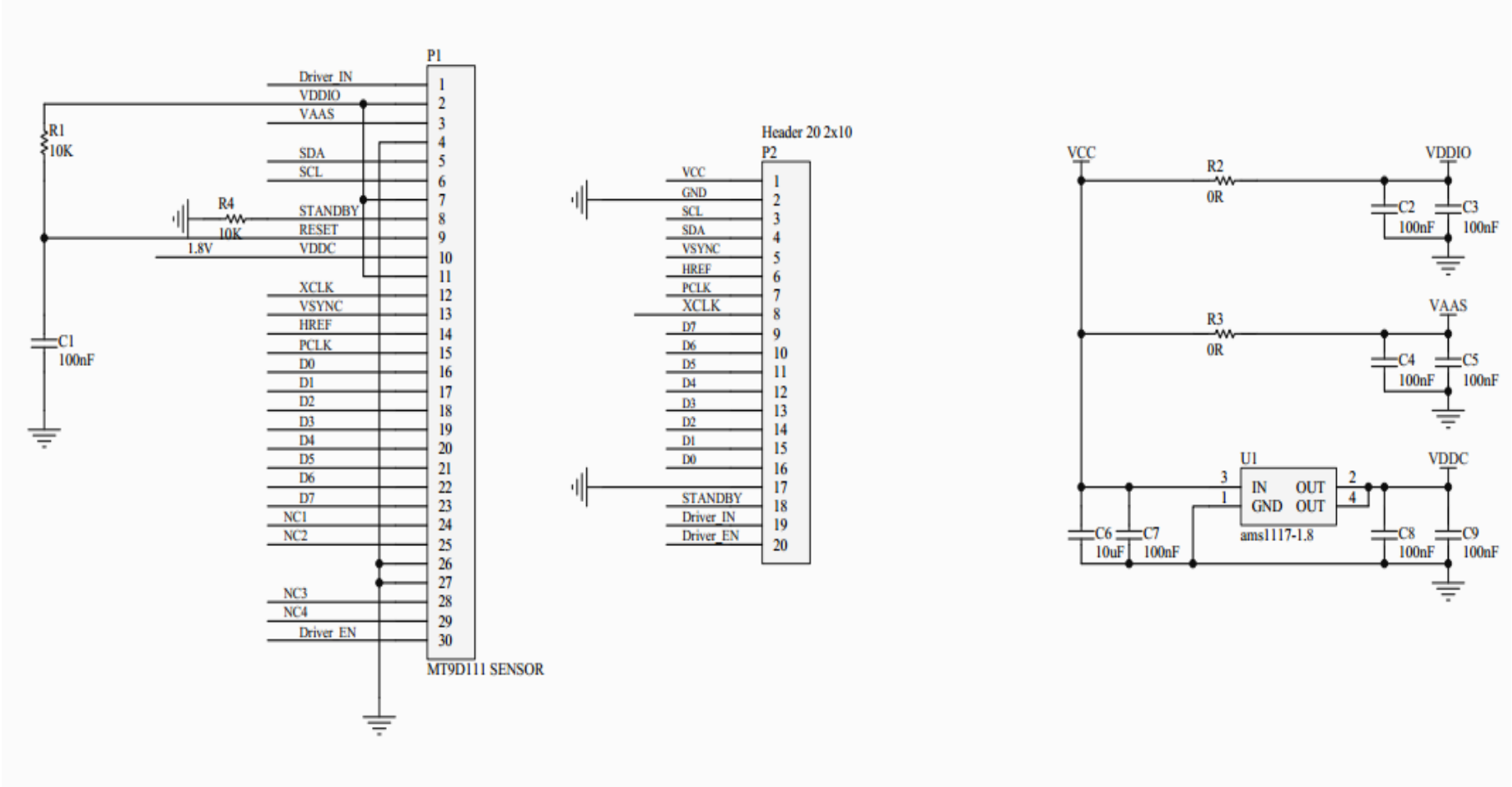

Figure 24 – MT9D111 Schematic

4.4.1.3 Mapping cc3200 Launchpad to Mt9d111

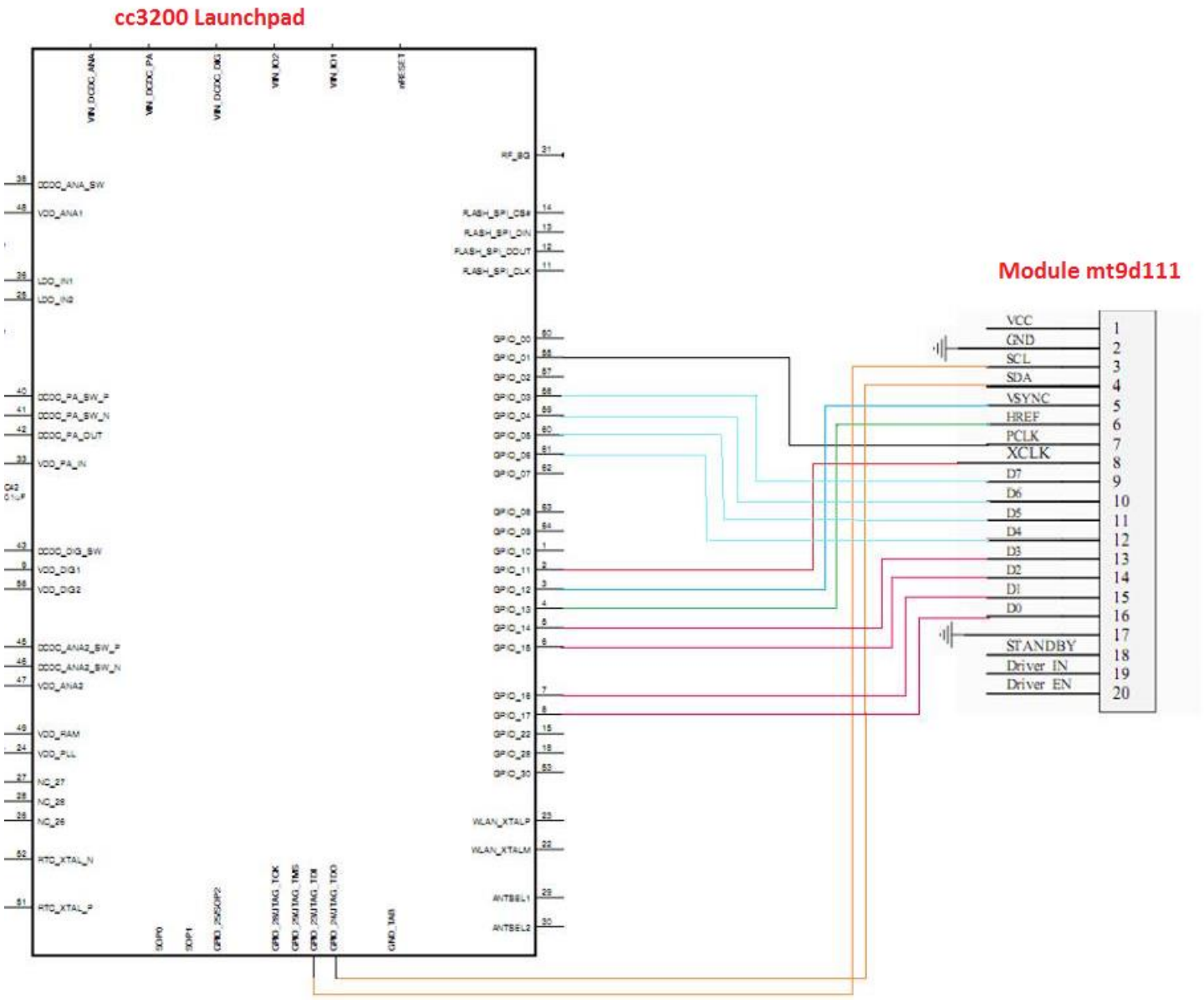

Figure 12 – Mapping Launchpad to Mt9d111

4.4.2 Software

4.4.2.1 SC Server Class Diagram

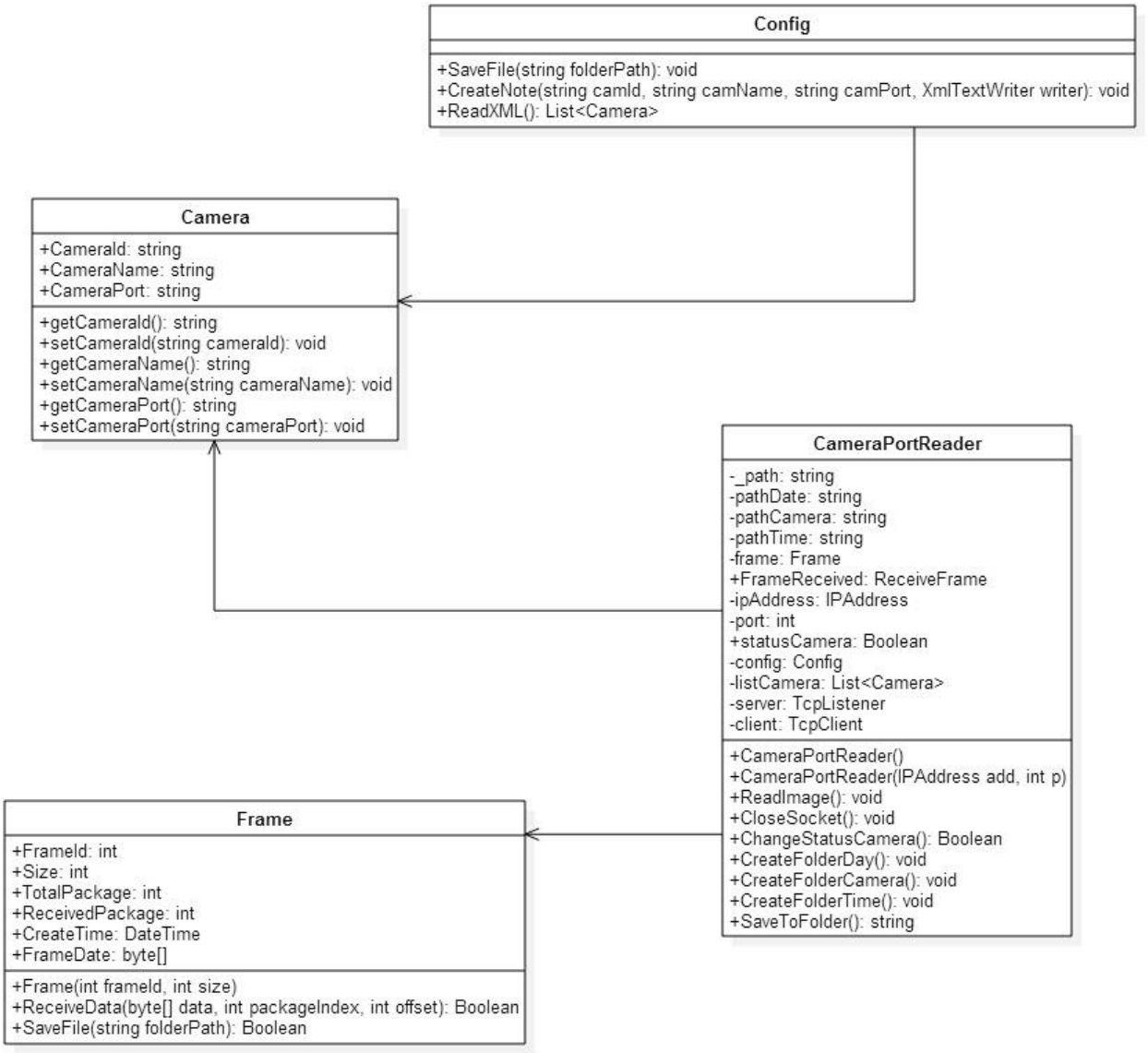

Figure 13 – SC Server Class Diagram

4.4.2.2 SC Server Class Diagram Explanation

4.4.2.2.1 Config

Methods

Methods	Type	Visibility	Description
CreateFileXml	Void	Public	Create file xml
CreateNote	Void	Public	Create information xml
SavetoXml	Void	Public	Save information to xml
ReadXML	List<Camera>	Public	List camera

4.4.2.2.2 Camera

Attributes

Attributes	Type	Visibility	Description
CameraId	String	Public	Id camera
CameraName	String	Public	Camera name
CameraPort	String	Public	Camera port
CameraDes	String	Public	Camera description

4.4.2.2.3 CameraPortReader

Attributes

Attributes	Type	Visibility	Description
_path	string	private	Address folder camera
pathDate	String	private	pathDate
pathCamera	String	private	pathCamera
pathTime	String	private	pathTime
Frame	Frame	Private	Class Frame
ipAddress	IPaddress	Private	ipAddress

Port	Int	Private	Port number
statusCamera	Boolean	Public	Status camera
Config	Config	Private	Class config
listCamera	List<Camera>	Private	List Camera

Methods

Methods	Type	Visibility	Description
ReadImage	Void	Public	Read image
CloseSocket	Void	Public	Close Socket
ChangeStatussCamera	Boolean	Public	Change status camera
CreateFolderDay	void	Public	Create folder day
CreateFolderCamera	void	Public	Create folder camera
CreateFolderTime	void	Public	Create folder time
SavetoFolder	Void	Public	Save image to folder

4.4.2.2.4 Frame

Attributes

Attributes	Type	Visibility	Description
FrameId	Int	Public	Frameid
Size	Int	Public	Size Frame
TotalPackage	Int	Public	Total package
CreateTime	Datetime	Public	Create datetime now
FrameData	Byte[]	Public	Frame data

Methods

Methods	Type	Visibility	Description
ReceiveData	Boolean	Public	Copy data from camera
SaveFile	Boolean	Public	Save image file to folder

4.4.2.3 Surveillance Camera Class Diagram

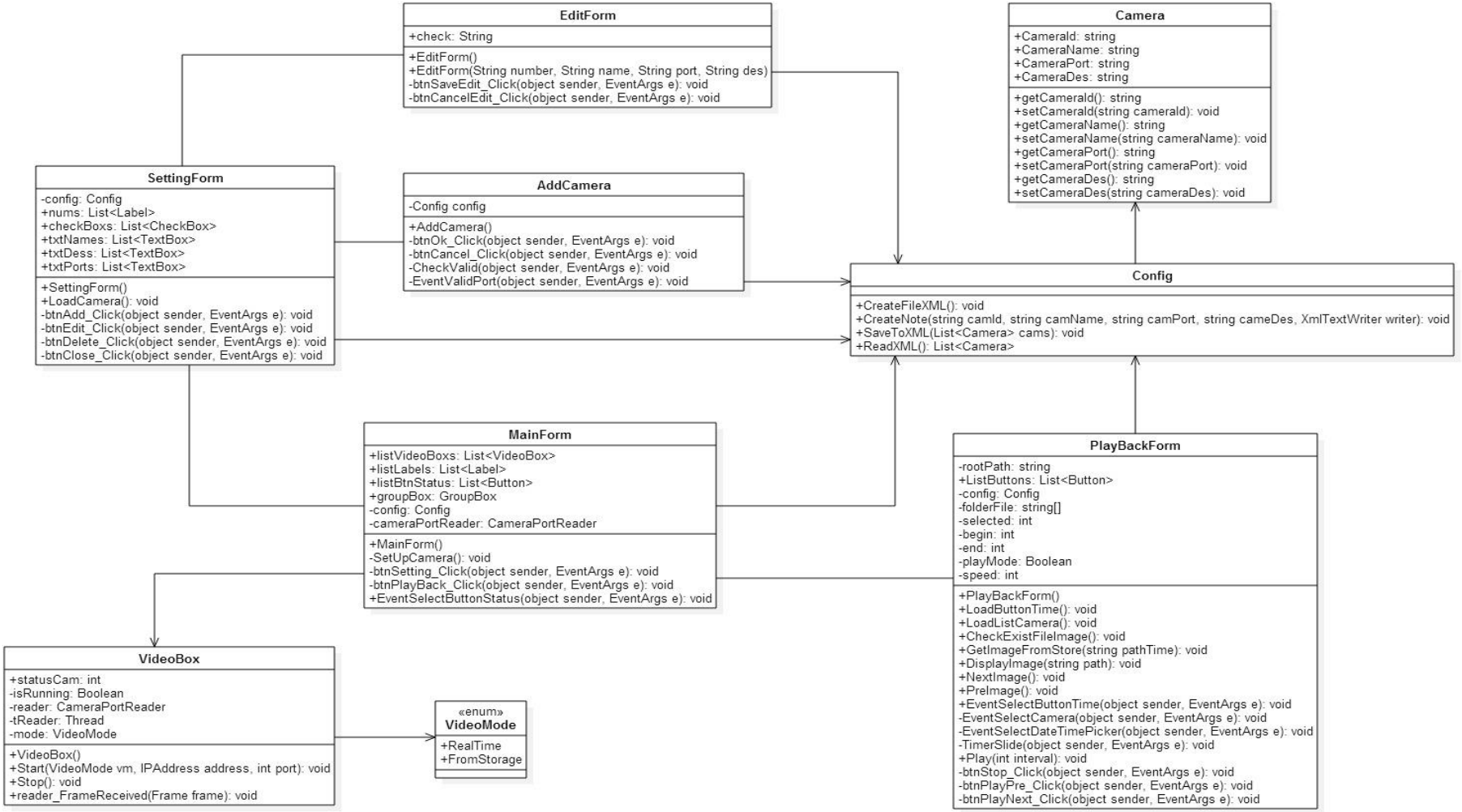

Figure 14 – Surveillance Camera Class Diagram

4.4.2.4 SC Class Diagram Explanation

4.4.2.4.1 MainForm

Attributes

Attributes	Type	Visibility	Description
listVideobox	List<Videobox>	Public	List Videobox
listLabels	List<Label>	Public	List Labels
listBtnStatus	List<Button>	Public	List Button
Config	Config	Private	class config
cameraPortReader	cameraPortReader	Private	Camera PortReader

Methods

Methods	Type	Visibility	Description
setupCamera	Void	Private	Setup to Camera
btnSetting_click	Void	Private	Show Setting Form
btnPlayback_click	Void	Private	Show Playback Form
EventSelectButtonStatus	Void	Public	Display event button

4.4.2.4.2 AddCamera

Attributes

Attributes	Type	Visibility	Description
Config	Config	Private	Class config

Methods

Methods	Type	Visibility	Description
btnOk_click	Void	Private	Confirm to add Camera
btnCancel_click	Void	Private	Cancel add Camera
EventValidPort	Void	Public	Check Valid Port

4.4.2.4.3 PlaybackForm

Attributes

Attributes	Type	Visibility	Description
rootPath	String	Private	Address folder camera
Config	Config	Private	Class config
listButton	List<Button>	Public	List Button
folderFile	String	Private	Folder file
Selected	Int	Private	Choose image in folder
Speed	Int	Private	Load speed image

Methods

Methods	Type	Visibility	Description
setupCamera	Void	Private	Stop to Camera
btnStop_click	Void	Private	Stop playback
btnPlayPre_click	Void	Private	Deceleration play video
btnPlayNext_click	Void	Private	Acceleration play video
LoadButtonTime	Void	Public	Load button in form
loadListCamera	Void	Public	Load list camera in form
CheckExistFileImage	Void	Public	Check file image
GetImageFromStore	Void	Public	Get file image
DisplayImage	Void	Public	Display Image
NextImage	Void	Public	Next Image
PreImage	Void	Public	Previous Image

4.4.2.4.4 EditFrom

Attributes

Attributes	Type	Visibility	Description
Check	String	Public	Check data in xml

Methods

Methods	Type	Visibility	Description
btnSaveEdit_click	Void	Private	Save data camera
btnCancelEdit_click	Void	Private	Cancel edit camera

4.4.2.4.5 Camera

Attributes

Attributes	Type	Visibility	Description
CameraId	String	Public	Id camera
CameraName	String	Public	Camera name
CameraPort	String	Public	Camera port
CameraDes	String	Public	Camera description

4.4.2.4.6 Config

Methods

Methods	Type	Visibility	Description
CreateFileXml	Void	Public	Create file xml
CreateNote	Void	Public	Create information xml
SavetoXml	Void	Public	Save information to xml
ReadXML	List<Camera>	Public	List camera

4.4.2.4.7 SettingFrom

Attributes

Attributes	Type	Visibility	Description
Config	Config	Private	Class config
checkBoxs	List<checkbox>	Public	List checkbox
txtNames	List<TextBox>	Public	List camera name
txtDes	List< TextBox >	Public	List camera description
textPort	List< TextBox >	Public	List camera port

Methods

Methods	Type	Visibility	Description
btnAdd_click	Void	Private	Add new camera to xml
btnEdit_click	Void	Private	Edit camera to xml
btnDelete_click	Void	Private	Delete camera to xml
btnClose_click	Void	Private	Close form

4.4.2.4.8 Videobox

Attributes

Attributes	Type	Visibility	Description
statusCam	Int	Public	Status camera
isRunning	Boolean	Private	Check videobox

Methods

Methods	Type	Visibility	Description
Start	Void	Public	Start video box
Stop	Void	Public	Stop video box
reader_FrameReceived	Void	Public	Read frame image

4.5 Flow Chart

4.5.1 Main Flow

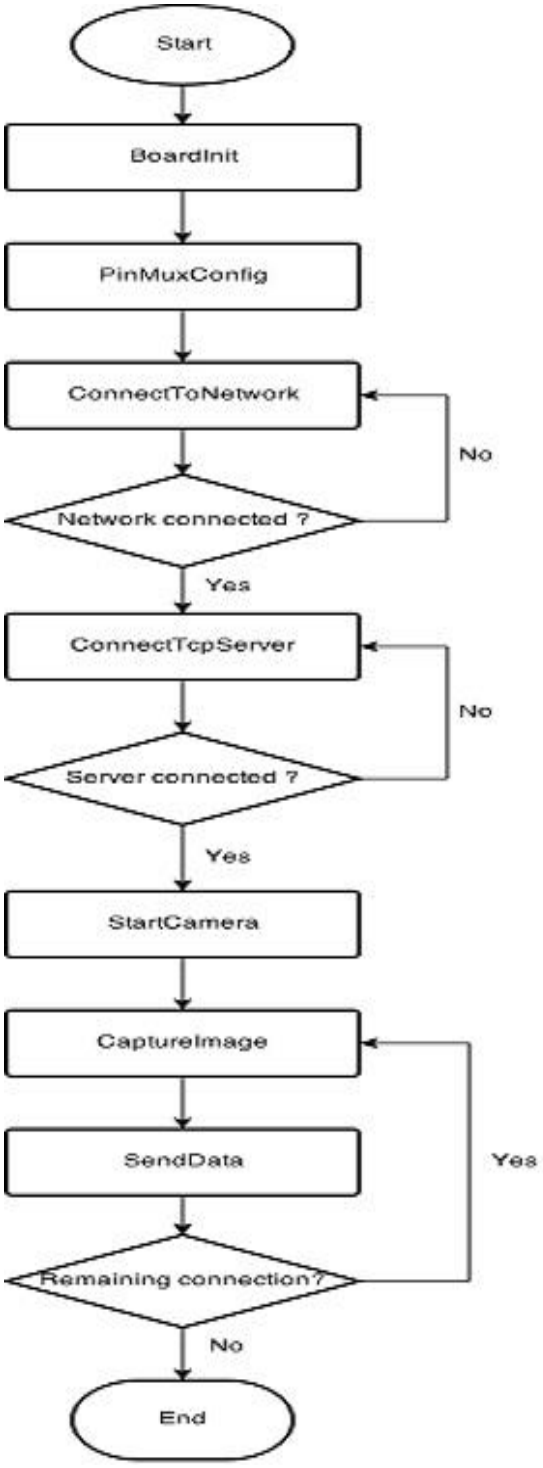

Figure 15 – Main Flowchart

4.5.2 Init Camera

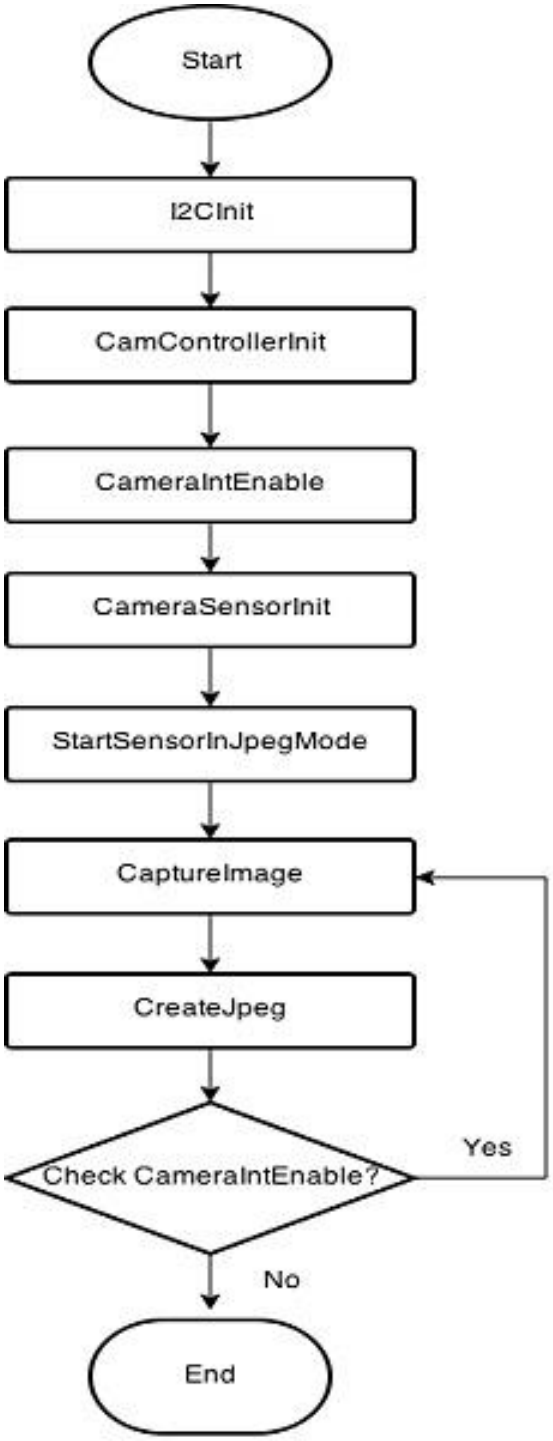

Figure 16 – Init Camera Flowchart

4.5.3 Send Package

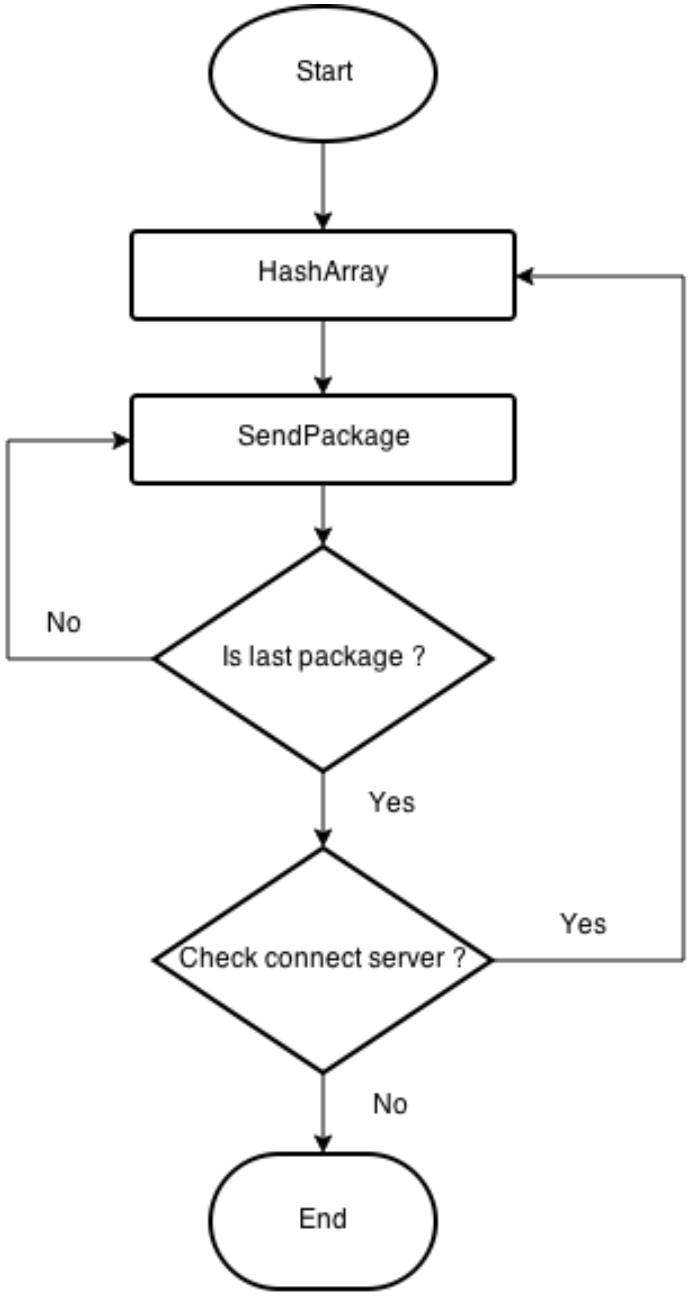

Figure 30 – Send Package Flowchart

4.5.4 SmartConfig

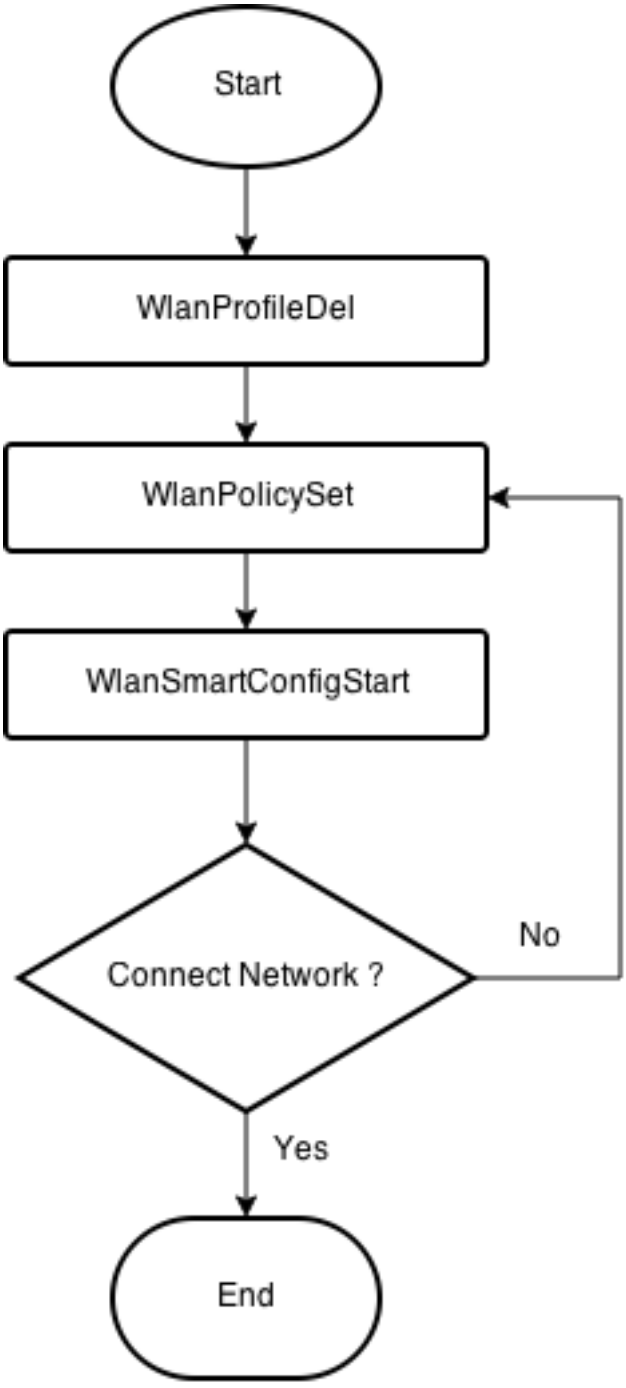

Figure 17 – SmartConfig Flowchart

4.6 Sequence Diagram

4.6.1 View camera

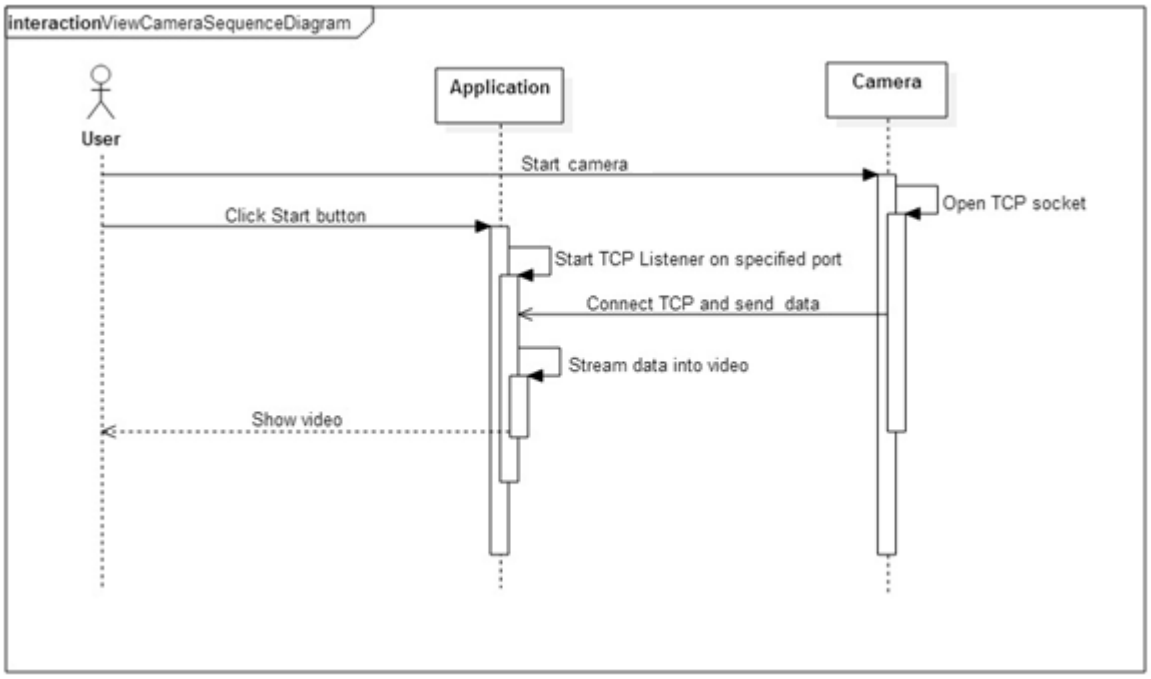

Figure 18 – View Camera Sequence Diagram

4.6.2 SmartConfig

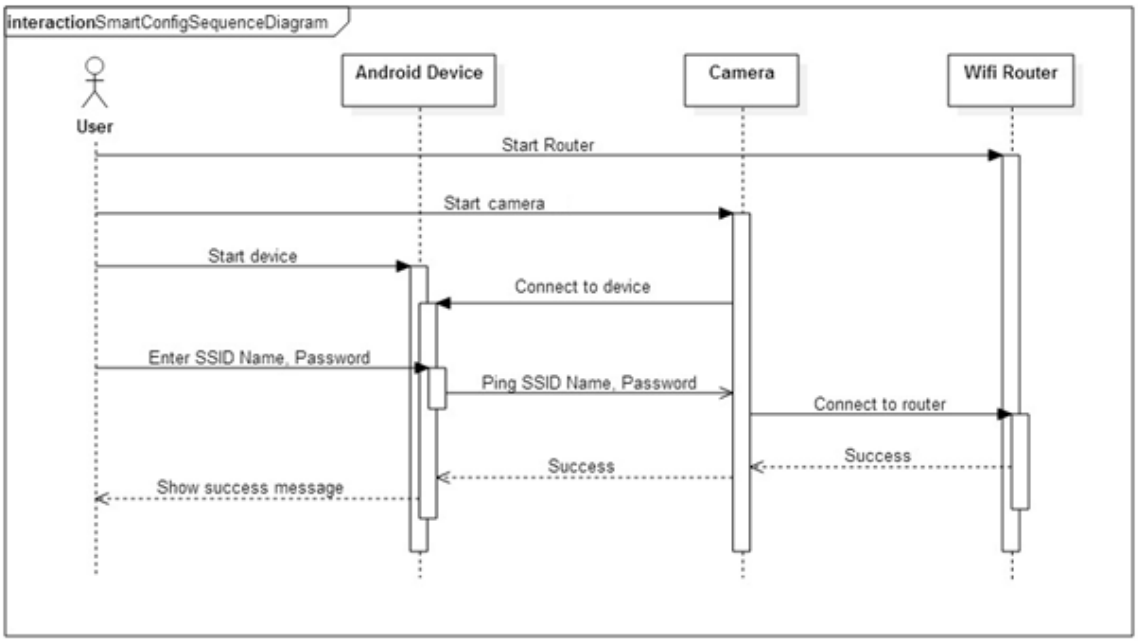

Figure 19 – SmartConfig Sequence Diagram

4.7 User Interface Design/Hardware Interface Design

4.7.1 Main Form

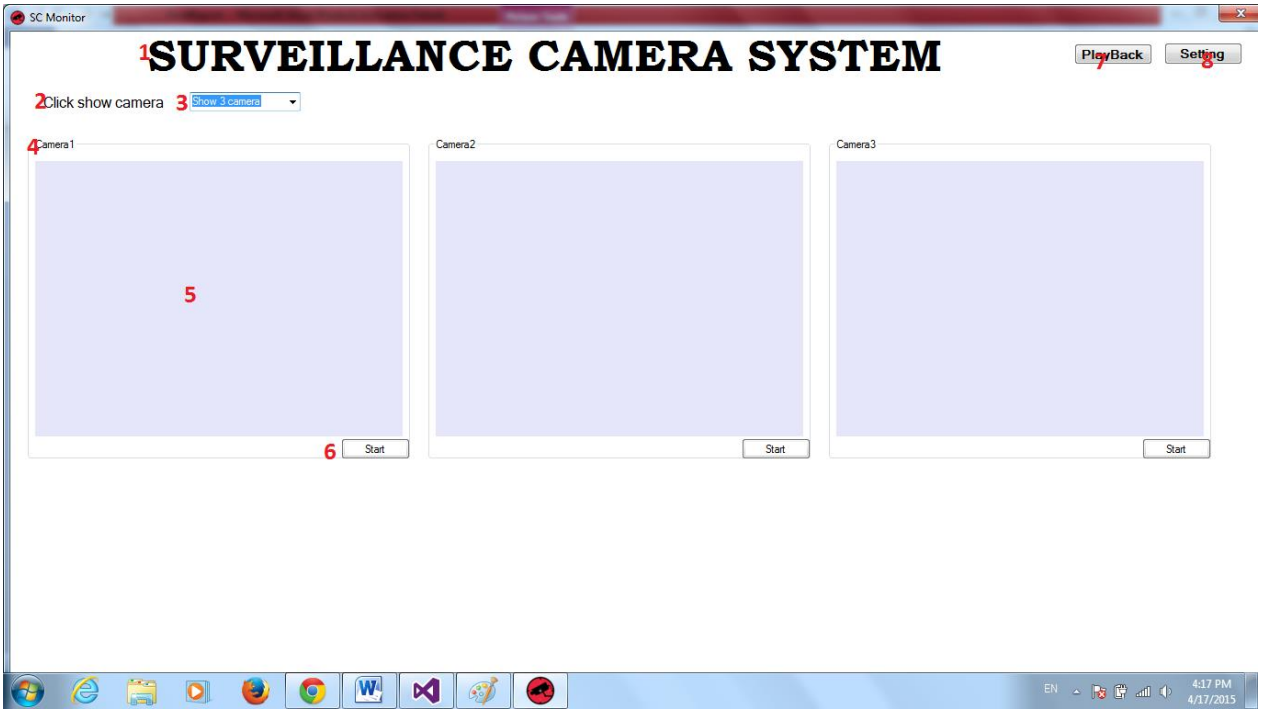

Figure 20 – SC Monitor Main Form

No	Item	Description
1	Label	Title “SURVEILLANCE CAMERA SYSTEM”
2	Label	Title “Click show camera”
3	Combobox	Display of camera
4	Label	Camera name
5	VideoBox	Show camera image
6	Button	Start/Stop camera connection
7	Button	Open Playback Form
8	Button	Open Setting Form

Table 17 – SC Monitor Main Form Description

4.7.2 Setting Form

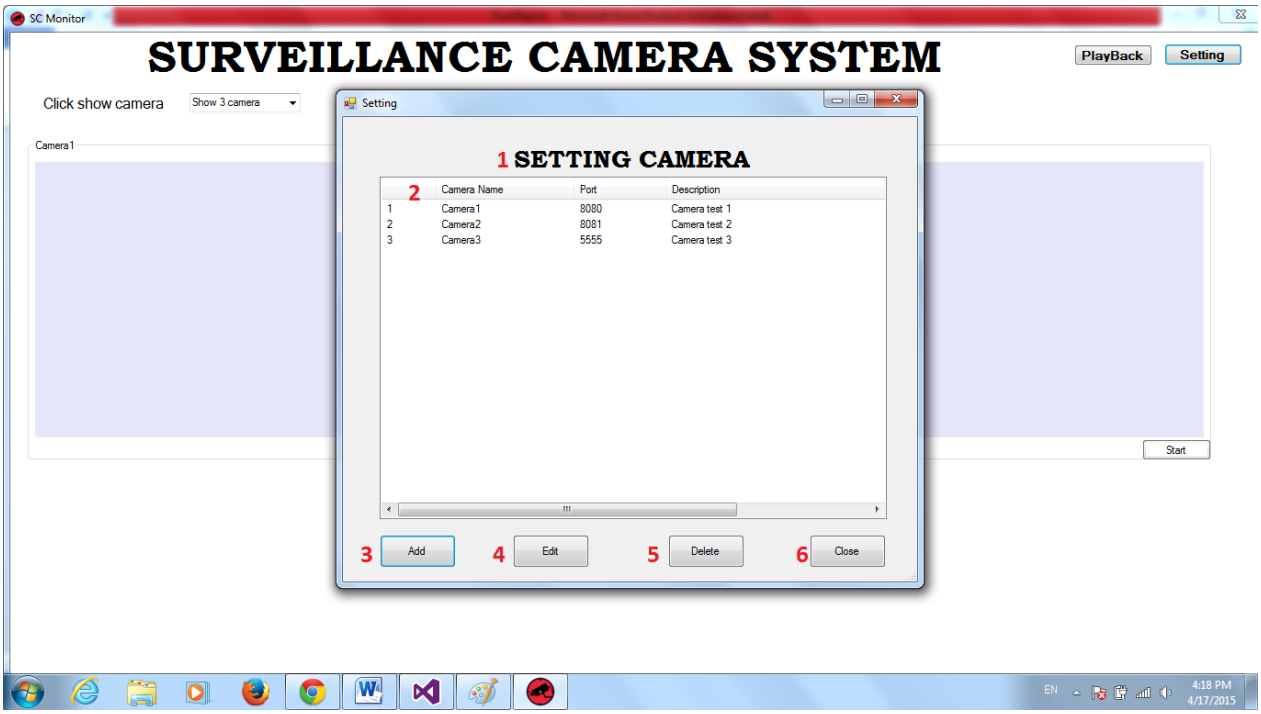

Figure 21 –Setting Form

No	Item	Description
1	Label	Title “SETTING CAMERA”
2	ListView	Show information of added cameras
3	Button	Open Add Camera Form
4	Button	Open Edit Camera Form
5	Button	Delete selected camera
6	Button	Close Setting Form

Table 18 – Setting Form Description

4.7.2.1 Add Camera Form

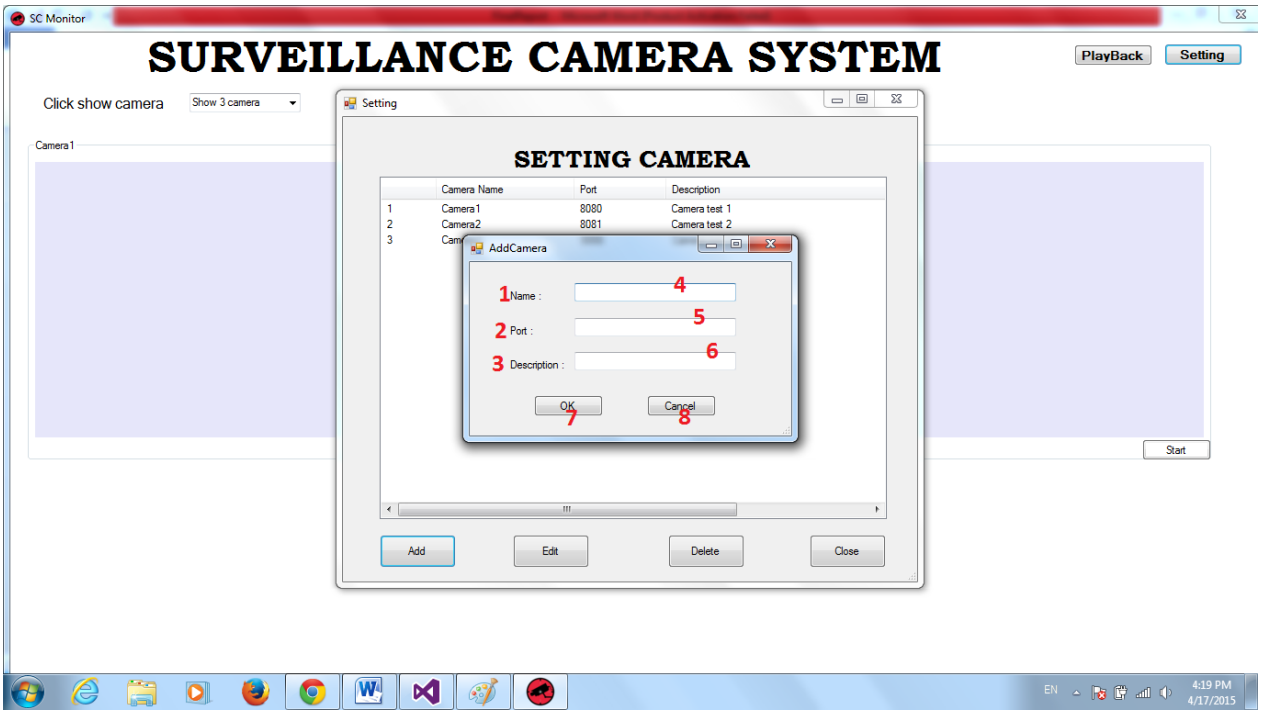

Figure 22 – Add Camera Form

No	Item	Description
1	Label	"Name" label
2	Label	"Port" label
3	Label	"Description" label
4	TextBox	Enter camera name
5	TextBox	Enter camera port to connect
6	TextBox	Enter description of camera
7	Button	Add new camera based on information above
8	Button	Cancel adding camera and close form

Table 19 – Add Camera Form Description

4.7.2.2 Edit Camera Form

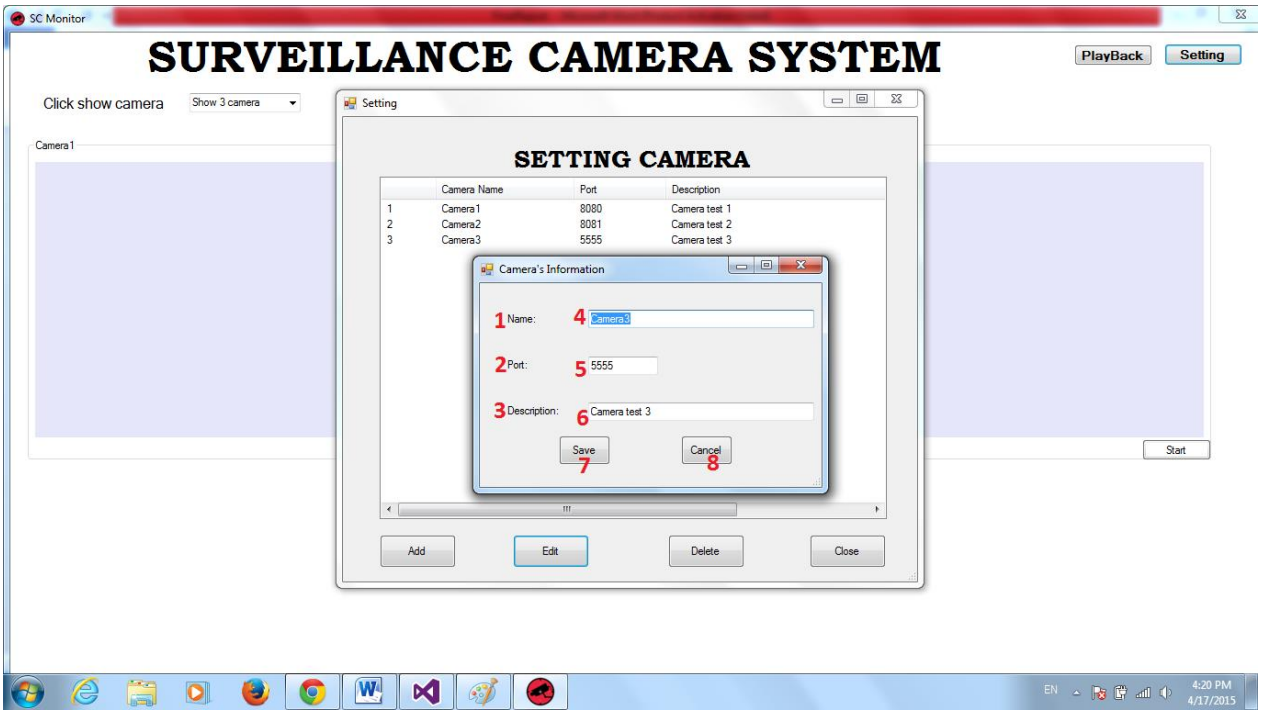

Figure 37–Edit Camera Form

No	Item	Description
1	Label	"Name" label
2	Label	"Port" label
3	Label	"Description" label
4	TextBox	Name of selected camera, enter new value to change
5	TextBox	Port of selected camera, enter new value to change
6	TextBox	Description of selected camera, enter new value to change
7	Button	Save new information of camera to XML file
8	Button	Cancel editing camera and close form

Table 20 – Edit Camera Form Description

4.7.3 Playback Form

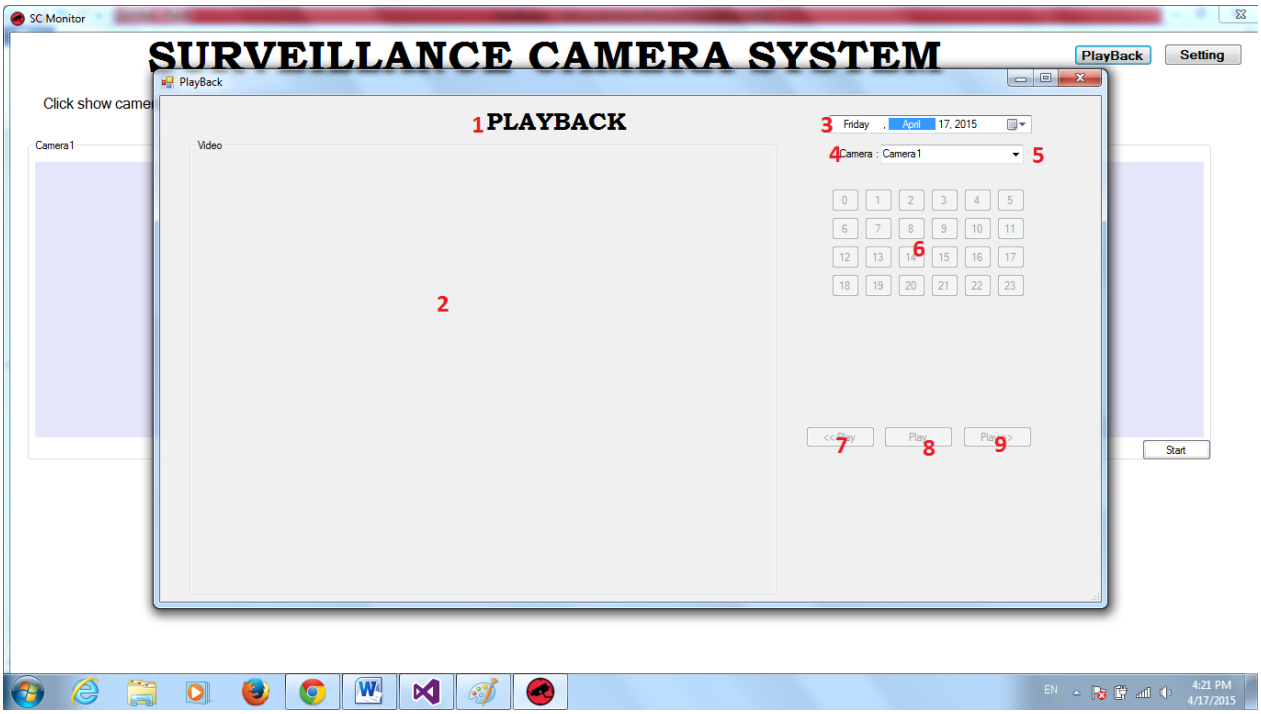

Figure 23 – Playback Form

No	Item	Description
1	Label	Title "PLAYBACK"
2	PictureBox	Show recorded video
3	DateTimePicker	Choose date of recorded video
4	Label	"Name" label
5	ComboBox	Choose camera name to view
6	Button	Choose hour of the recorded video in specified date, red button means that hour has recorded video
7	Button	Playing video backward with 1x 2x 4x speed
8	Button	Play/Stop recorded video
9	Button	Playing video forward with 1x 2x 4x speed

Table 21 – Playback Form Description

4.7.4 SmartConfig

4.7.4.1 SmartConfig Tab

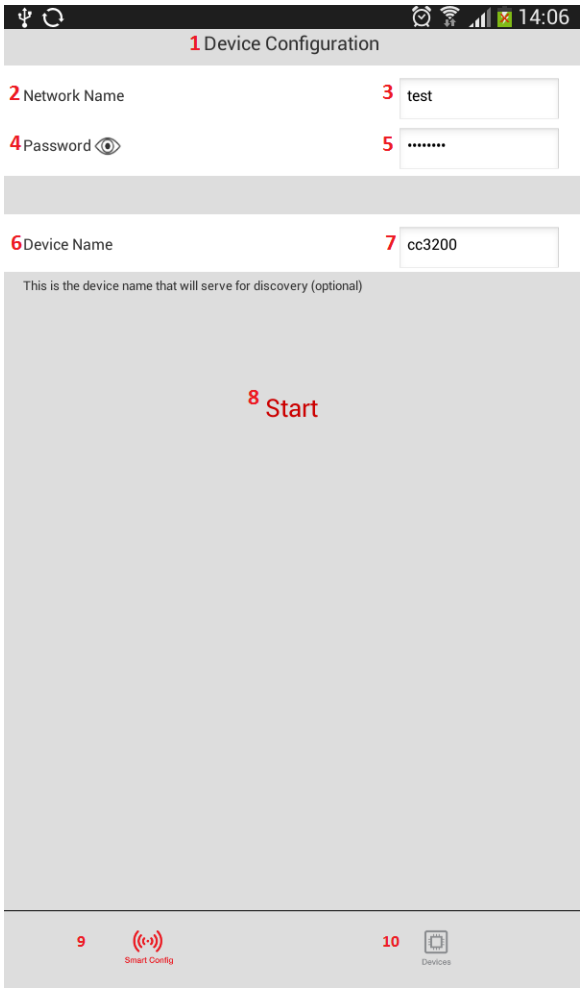

Figure 24 –SAA SmartConfig Tab

No	Item	Description
1	Label	Title “Device Configuration”
2	Label	“Network Name” label
3	EditText	Enter network name
4	Label	“Password” label
5	Edit Text	Enter password of the network
6	Label	“Device Name” label

7	EditText	Enter device name to configure
8	Button	Start device configuration based on information above
9	Tab	Change to SmartConfig tab
10	Tab	Change to Devices tab

Table 22 – SAA SmartConfig Tab Description

4.7.4.2 Devices Tab

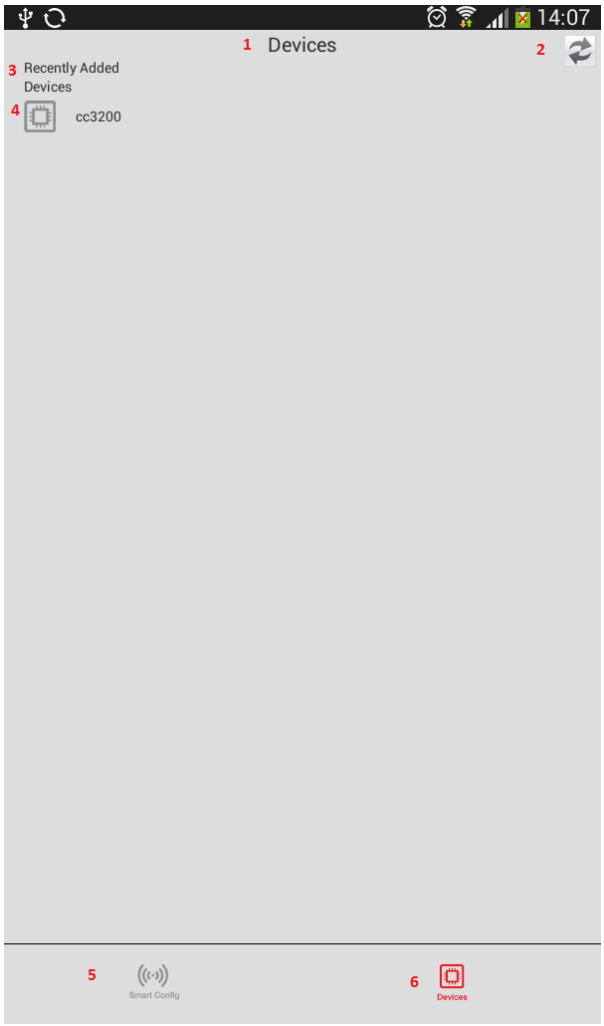

Figure 40 –SAA Devices Tab

No	Item	Description
1	Label	Title “Devices”

2	Button	Refresh added device list
3	Label	“Recently added devices” label
4	Item	Show added devices
5	Tab	Change to SmartConfig tab
6	Tab	Change to Devices tab

Table 23 –SAA Devices Tab Description

4.8 Camera Database Design

- Information coded in XML is easy to read and understand, plus it can be processed easily by computers. Tags, attributes and element structure provide context information ... opening up new possibilities for highly efficient search engines, intelligent data mining, agents, etc.

- It is a textual data format with strong support via Unicode for different human languages. It can adhere to a concrete schema if so you choose. Once it validates, you can say it's correct and start parsing, near ubiquitous support in a wide array of languages and frameworks.

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>
- <Table>
  - <Camera>
      <Id>1</Id>
      <Name>Camera1</Name>
      <Port>8080</Port>
      <Description>Camera test 1</Description>
    </Camera>
  - <Camera>
      <Id>2</Id>
      <Name>Camera2</Name>
      <Port>8081</Port>
      <Description>Camera test 2</Description>
    </Camera>
  - <Camera>
      <Id>3</Id>
      <Name>Camera3</Name>
      <Port>5555</Port>
      <Description>Camera test 3</Description>
    </Camera>
  - <Camera>
      <Id>4</Id>
      <Name>Camera4</Name>
      <Port>1232</Port>
      <Description>adw</Description>
    </Camera>
</Table>
```

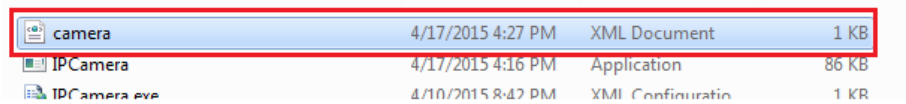

Figure 41 –Camera Database Design

5. SYSTEM IMPLEMENTATION & TEST (SIT)

5.1 Introduction

5.1.1 System Overview

In this section, all necessary information about test plan, test cases, test result, the environment for testing and test pass/fail criteria are provided, as well as risks estimations and a checklist for testing this system.

5.1.2 Test Approach

- **Goal**: To test the whole system based on the core workflow.
- **Method**: Black-box Testing.

The purpose of this section is to verify and ensure that SC's function meets its design specification and other requirements from user. The following part will describe which features to be tested.

5.1.3 Test plan

We have two main test phases: Function test (Integration test) and System test.

For Integration test, we use the functional test. This kind of test shows us the working of each function right or wrong. We focus on the result of the function in different cases.

The System test check all the activities of the camera after setup such as how it works and if something happen, what it will do. We create test cases to check each activity of the cameras from turn on to turn off. Then, we observe and record the result.

5.1.4 Test past/fail criteria

For system testing, the criteria are:

- 90% of the test cases must pass.
- 100% of test cases about hardware module must pass.
- All test cases dealing with critical functionality must pass.
- All medium and high severity defects must be fixed.

- Test coverage must be at least 90%.

5.1.5 Test environment

Test tool:

- 2 PCs with Windows 7 Operating System, which connected to the same network with SC.
- An Android smartphone.
- Visual studio 2013 Community.

5.2 Test Case

5.2.1 Surveillance Camera (SC) Test Cases

No.	Function			Pre-condition	Step of testing	Expected result	Create by	Executed by	Result
	Large function	Medium function	Small function						
1	Camera	Connect network	Connect to a network	1. Camera has been turned on.	1. Connect Android device to a network 2. Open SAA. 3. Enter name and pass work of the network, also enter the name of the camera. 4. Click on “Start”	1. New camera is added to the device list in SAA. 2. Camera is connected to the network. 3. State: data red led in camera is blinking..	DuyNT	DuyNT	Pass

2		Send video	Send video	1. Camera has been turned on. 2. Camera has connected to the network.	1. Connect computer to the network. 2. Open SCM on computer. 3. Add the camera with a specific port number in order to transmit the video via network connection. 4. Start the connection between camera and computer.	1. State: video plays for 1 hour without errors. 2. State: data red led in camera is blinking.	DuyNT	DuyNT	Pass

3		Send video	Handling picture	1. Camera has been turned on. 2. Camera has connected to the network. 3. Camera capture 4. Send video	1. Connect computer to the network. 2. Open Code Composer Studio 6.0.1 on computer. 3. Start the connection between camera and computer. 4. Run project and view the processing of camera	1. State: Camera handling frame rate stability. 2. State: Cc3200 handling 4 – 5 pictures per second	DuyNT	DuyNT	Pass

Table 24 – Test case for SC

5.2.2 SmartConfig Android App (SAA) Test Cases

No.	Function			Pre-condition	Step of testing	Expected result	Create by	Executed by	Result
	Large function	Medium function	Small function						
1	Connect cameras to network	Connect network	Connect to a network	1. Camera has been turn on. 2. Android device connected to the network.	1. Connect Android device to a network 2. Open SAA 3. Enter name and pass work of the network, also enter the name of the camera. 4. Click on “Start”	1. New camera(s) is added to the device list. 2. Camera connected to the network.	ThangHM	ThangHM	Pass

2			Connect to an unavailable network	1. Camera has been turn on. 2. Android device did not connect to the network.	1. Connect Android device to a network 2. Open SAA	1. A message box appears with message "No Wifi connection detected. What would you like to do?"	ThangHM	ThangHM	Pass
3			Connect cameras to the network with wrong password	1. Camera has been turn on. 2. Android device connected to the network.	1. Connect Android device to a network 2. Open SAA 3. Enter wrong name/password of the network. 4. Click on "Start".	1. A message box appears with message "No new device has been found"	ThangHM	ThangHM	Pass

4			Connect unavailable cameras to the network	1. Android device connected to the network.	1. Connect Android device to a network 2. Open SAA 3. Enter wrong name of the camera. 4. Click on "Start".	1. A message box appears with message "No new device has been found".	ThangHM	ThangHM	Pass

Table 25 – Test case for SAA

5.2.3 Surveillance Camera Monitor (SCM)Test Cases

No.	Function			Pre-condition	Step of testing	Expected result	Create by	Executed by	Result
	Large function	Medium function	Small function						

1	Camera	Receive video	First time connect to camera	1. Program was opened. 2. Camera has been turned on	1. Add a new camera. 2. Click on "Start" button.	1. A new camera box appears. 2. Video transmitted from the camera starts to play. 3. "Start" button change to "Stop".	ThangHM	AnhVQ	Pass
2			Stop receiving video	1. Connected to the network	1. Click on "Stop" button.	1. "Stop" button changes to "Start". 2. Video stops at the last received scene.	ThangHM	AnhVQ	
3			Re-receive the video	1. Camera connected to the network 2. Video freezes.	1. Click on "Start" button.	1. "Start" button changes to "Stop" 2. Video continues to play.	ThangHM	AnhVQ	Pass

4	Control cameras	Add camera	Add new camera	1. Program is opened.	1. Click on "Setting" button. 2. Click on "Add" button. 3. Enter camera information. 4. Click on "Add" button.	1. Camera is added to the list. 2. Camera box for the new camera is appear	DuyNT	AnhVQ	Pass

5		Edit camera	Edit information of a camera	1. Program is opened.	1. Click on "Setting" button. 2. Choose a camera. 3. Click on "Edit" button. 4. Enter camera information. 5. Click on "Save" button.	1. Information of the camera is changed in the list.	DuyNT	AnhVQ	Pass
6		Delete camera	Delete a camera from the list	1. Program is opened. 2. Camera(s) is added to the list.	1. Click on "Setting" button. 2. Choose a camera. 3. Click on "Delete" button. .	1. Camera is removed to the list. 2. Camera box of the chosen camera is deleted.	DuyNT	AnhVQ	Pass

7	View recorded videos	Control video	Play the video	Program is opened.	1. Click on "Playback" button. 2. Choose date. 3. Choose camera. 4. Choose hour. 5. Click on "Play" button.	1. The video starts to play. 2. "Play" button change to "Pause" button. 3. "<< Play" – play backward button become available. 4. "Play >>" – play forward button become available.	TuDVT	TuDVT	Pass
8			Pause the video	1. Program is opened. 2. Video is playing.	1. Click on "Pause" button.	1. The playing video is stopped. 2. "Pause" button change to "Play" button.	TuDVT	TuDVT	Pass

9			Play video with X2 speed	1. Program is opened. 2. Video is playing.	1. Click on "Play >>"button.	1. The video starts playing with X2 normal speed.	TuDVT	TuDVT	Pass
10			Play video with X4 speed	1. Program is opened. 2. Video is playing with X2 speed.	1. Click on "Play >>"button.	1. The video starts playing with X4 normal speed.	TuDVT	TuDVT	Pass
11			Play backward.	1. Program is opened. 2. Video is playing.	1. Click on "Pause" button. 2. Click on "<< Play" button.	1. The video starts playing backward.	AnhVQ	AnhVQ	Pass
12			Play backward with X2 speed	1. Program is opened. 2. Video is playing backward.	1. Click on "<< Play" button.	1. The video starts playing backward with X2 normal speed.	AnhVQ	AnhVQ	Pass

13			Play backward with X4 speed	1. Program is opened. 2. Video is playing backward with X2 speed.	1. Click on "<< Play" button.	1. The video starts playing backward with X4 normal speed.	AnhVQ	AnhVQ	Pass
14		View video	View video	1. Connected to the network 2. Start SCM	1. Click "Start" button	View video on SCM	AnhVQ	AnhVQ	Pass
15		Test processing	Processing video	1. Connected to the network 2. Start SCM	1. View video. 2. View picture in folder "Camera"	4-5 picture per second	AnhVQ	AnhVQ	Pass
16		Test picture	Test picture	1. Connected to the network 2. Start SCM	1. View picture in folder "Camera"	Size picture 640x480 pixel	AnhVQ	AnhVQ	Pass

Table 26 –Test Cases for SCM

5.2.4 Test Report

Test Report			
ProjectName	Surveillance Camera	**Creator**	ThangHM, DuyNT
ProjcetCode	SC	**Reviewer/Approver**	HungPD
DocumentCode	SC_TestReport_v1.0	**Issue Date**	
Note	All off module		

No	Module code	Pass	Fail	Untested	N/A	Number of testcase
1	Surveillance Camera	3	0	0	0	3
2	SmartConfig Android App	4	0	0	0	4
3	Surveillance Camera Monitor	16	0	0	0	16
Total		23	0	0	0	23
Test coverage				**100%**		
Test successful coverage				**100%**		

Table 27 – Test Report

5.3 Test Evident

5.3.1 Test SmartConfig Android App (SAA)

1. User open the application on Android device for choosing which Wi-Fi network that user want the camera to connect to.
2. User enter name and password of the Wi-Fi network that user want the camera to connect.
3. Click “Start” to run, user has to wait until app finish searching.

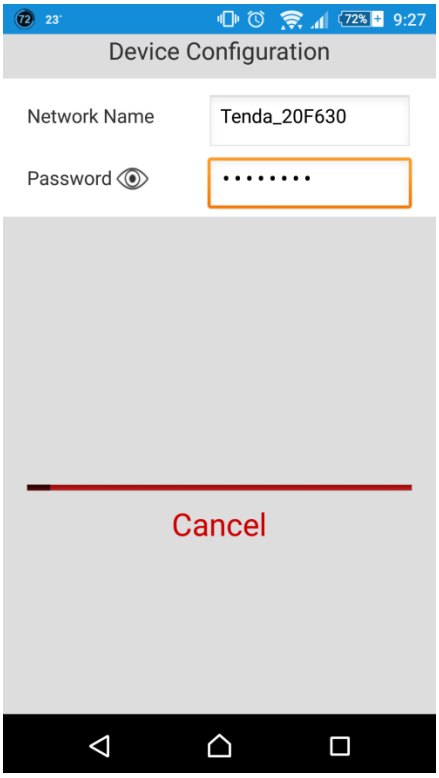

Figure 42 – SAA Searching

4. If searching result is not found, system will show error notification. If there is any searching result, user will find result keyword “MAC information cc3200”.

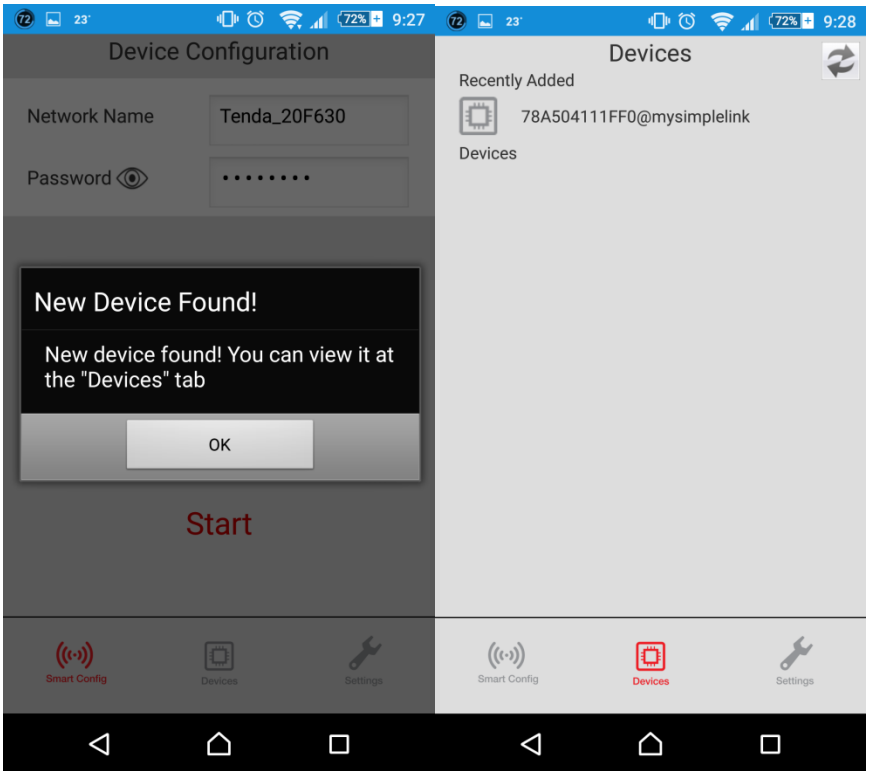

Figure 43 – SAA Show Result

5.3.2 Test Surveillance Camera (SC)

1. User open the Code Composer Studio 6.0.1(CCS) application, connect SC to application and executes it.
2. System will start SC and connect to network successful.

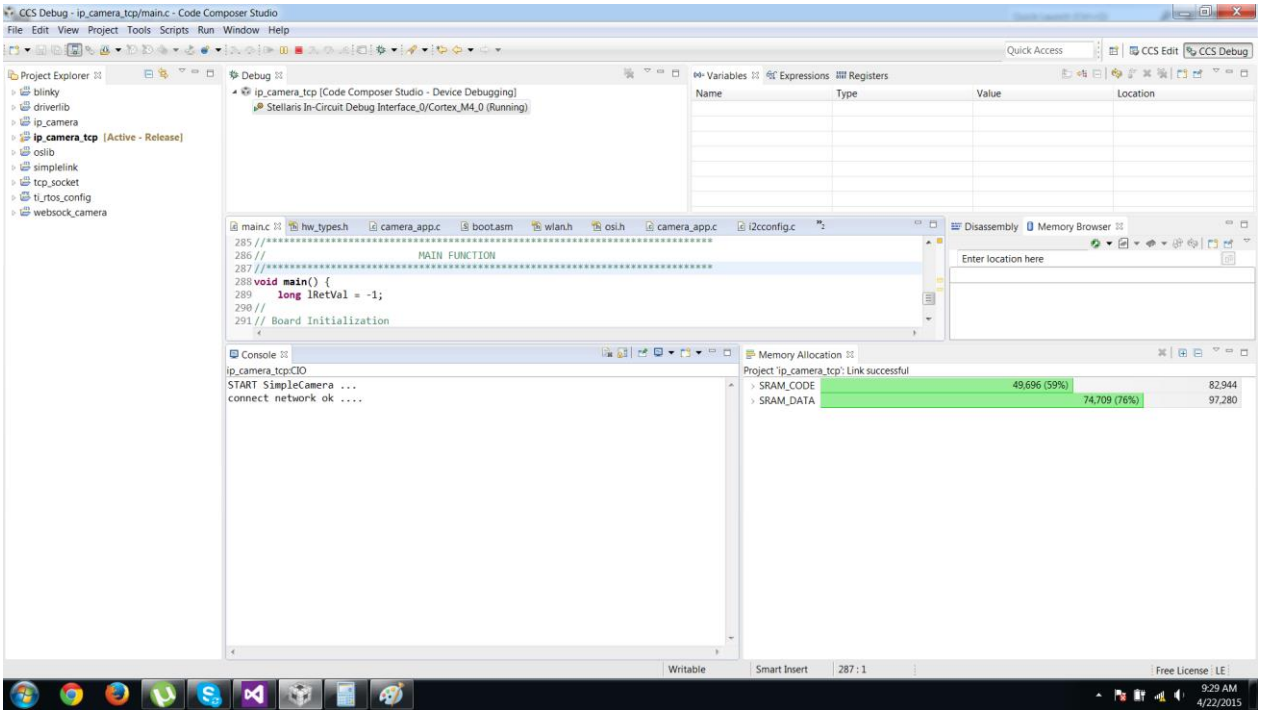

Figure 44 – SC Connect Successful

3. If connecting result, SC send data to SCM.

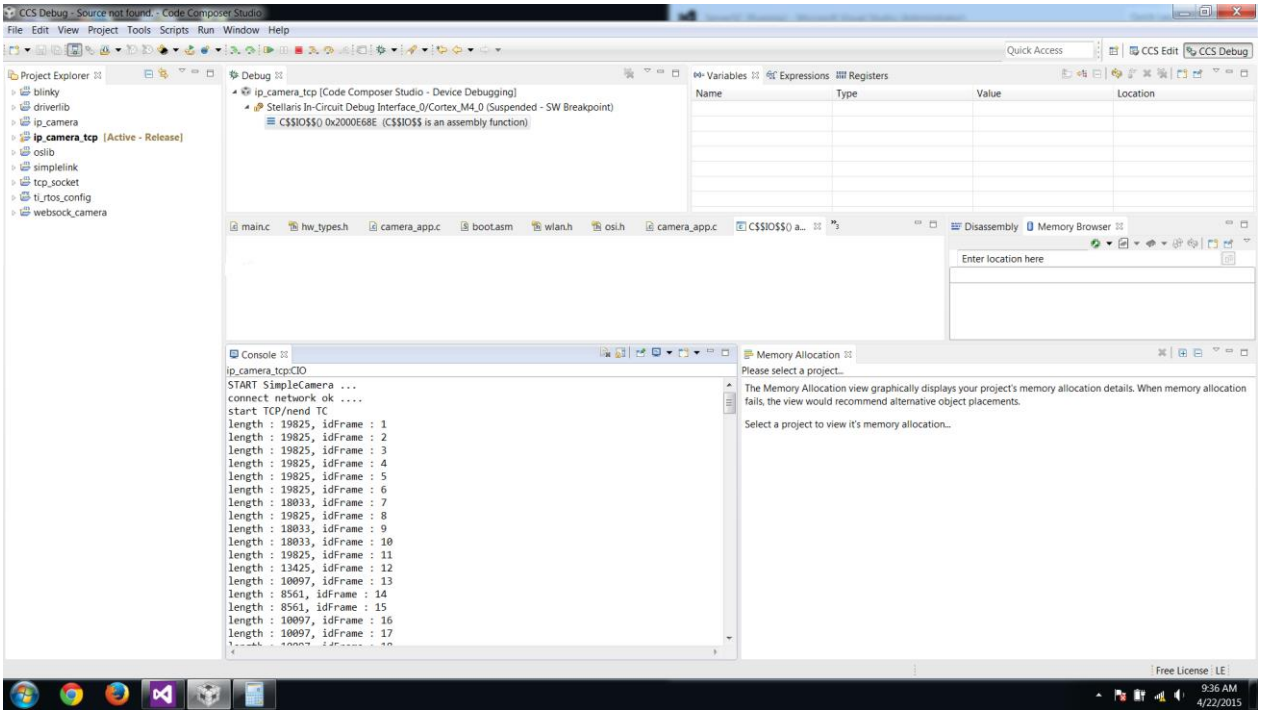

Figure 45 – SC Send Data

5.3.3 Test Surveillance Camera Monitor (SCM)

5.3.3.1 Test ViewVideo

1. User open the Visual Studio, choose SC project and executes it.
2. User can choose to display the number of cameras, and click “Start” camera.
3. After clicking on “Start” button, the computer starts receiving video signal from added port number of the camera and the video starts playing in the camera box.

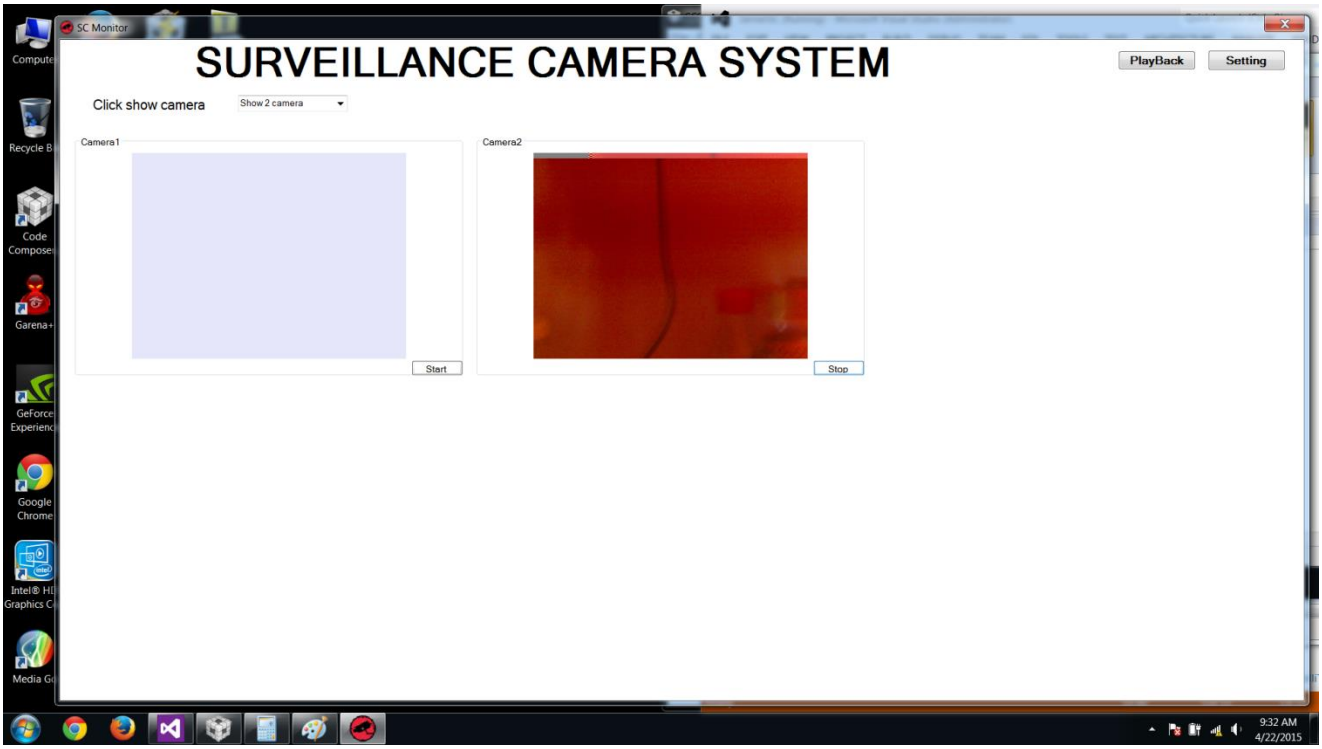

Figure 46 – SCM Test View Video

5.3.3.2 Test PlayBack

1. User open the Visual Studio, choose SC project and executes it..
2. User can click button “Playback”, then “PlaybackForm” is show.
3. In “PlaybackForm”, user can choose Calendar, Camera and Times ... Next, you click “Play” and see video.

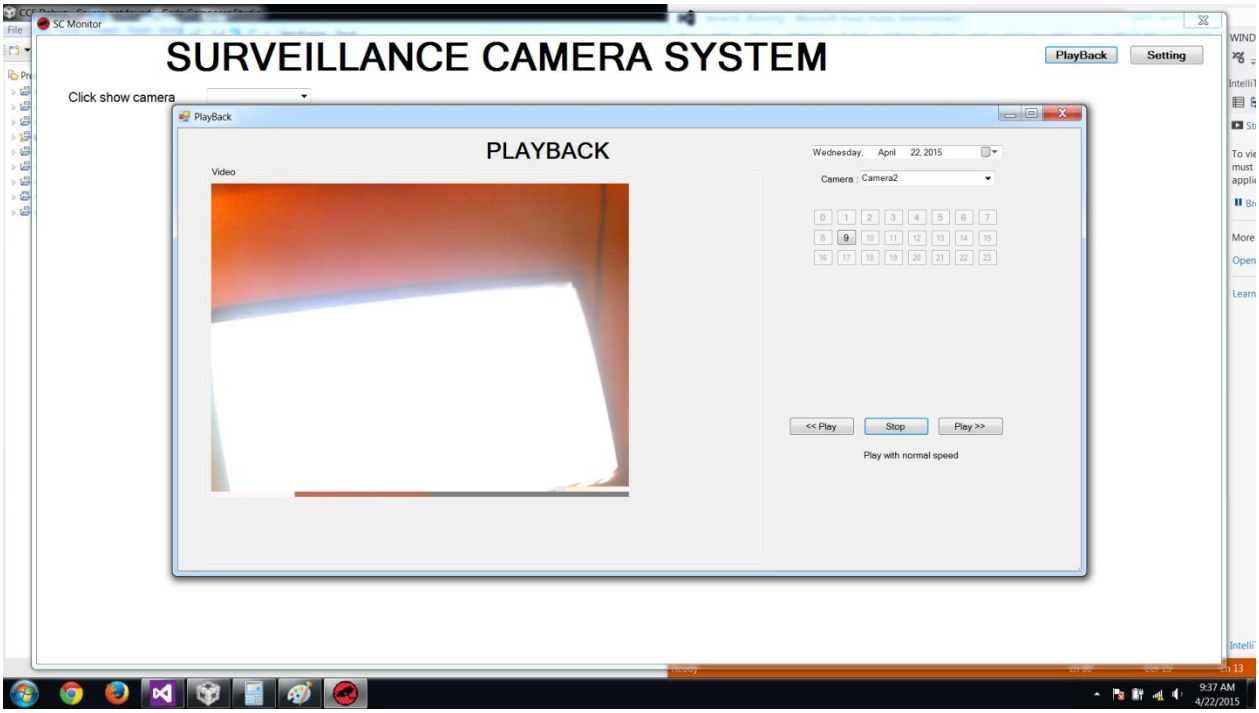

Figure 47 – SCM Test Playback

6. SYSTEM USER'S MANUAL (SUM)

6.1 System Requirement

CPU	Intel Pentium 4 2.4Ghz or higher
RAM	512MB or higher
OS	32bit or 64bit - Windows 7, 8, 8.1
Software	Microsoft .NET Framework 4 Client Profile or higher
Communication	Ethernet or Wi-Fi connection
Free disk space	15MB

Table 27 – System Requirement

6.2 SmartConfig Android App (SAA)

User must follow these steps to connect cameras to the specific network:

1. User open the application on Android device for choosing which Wi-Fi network that user want the camera to connect to.

Figure 48 – SAA Welcome Screen

2. User enter name and password of the Wi-Fi network that user want the camera to connect to.

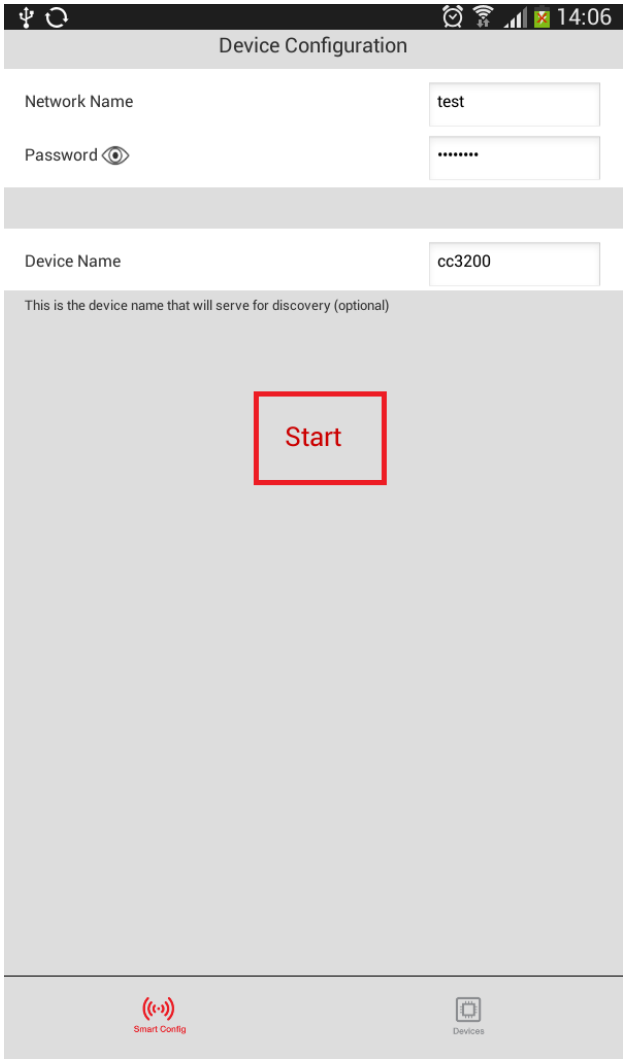

Figure 49 – Start connect camera to the network

3. The application searches if there is any camera match the entered name, then it will sends a command to the camera to connect to the Wi-Fi network using the entered information

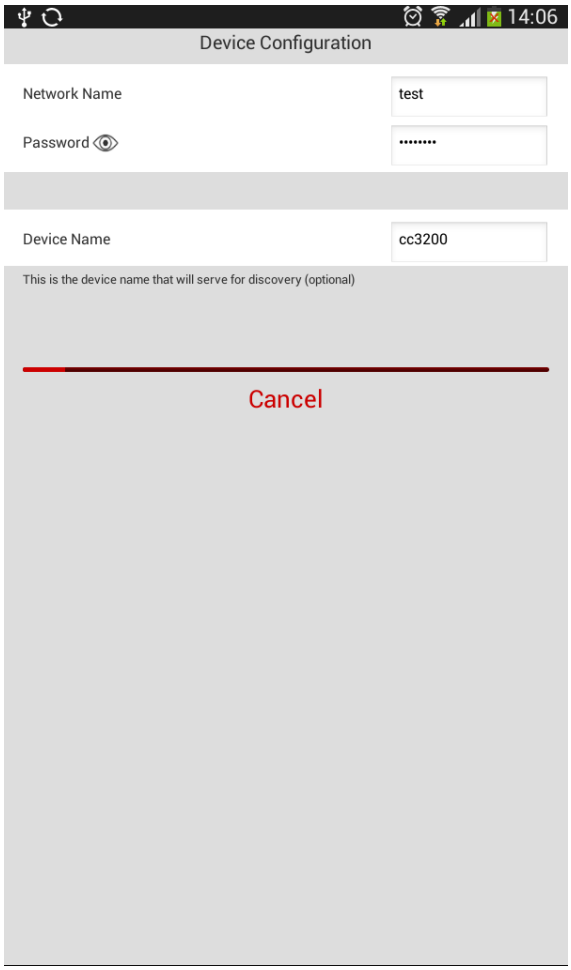

Figure 50 – Searching for matched cameras

4. After the searching is completed, the application will change to Devices tab with the list of cameras.

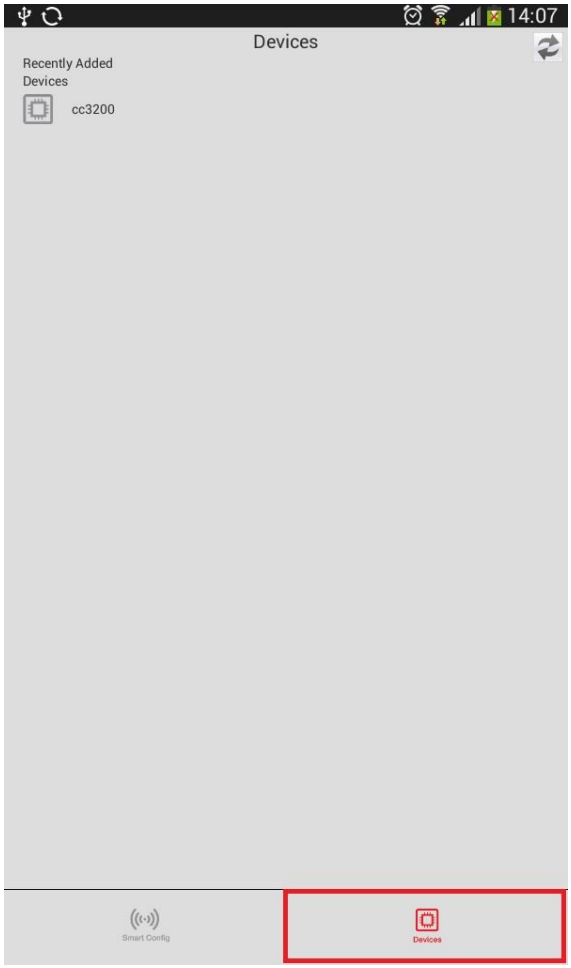

Figure 51 – Connect camera to the network succesful

6.3 Surveillance Camera Monitor (SCM)

6.3.1 Watching videos

User can choose to display the number of cameras.

Next, user can start watching videos from specific cameras by clicking on "Start" button below every camera box.

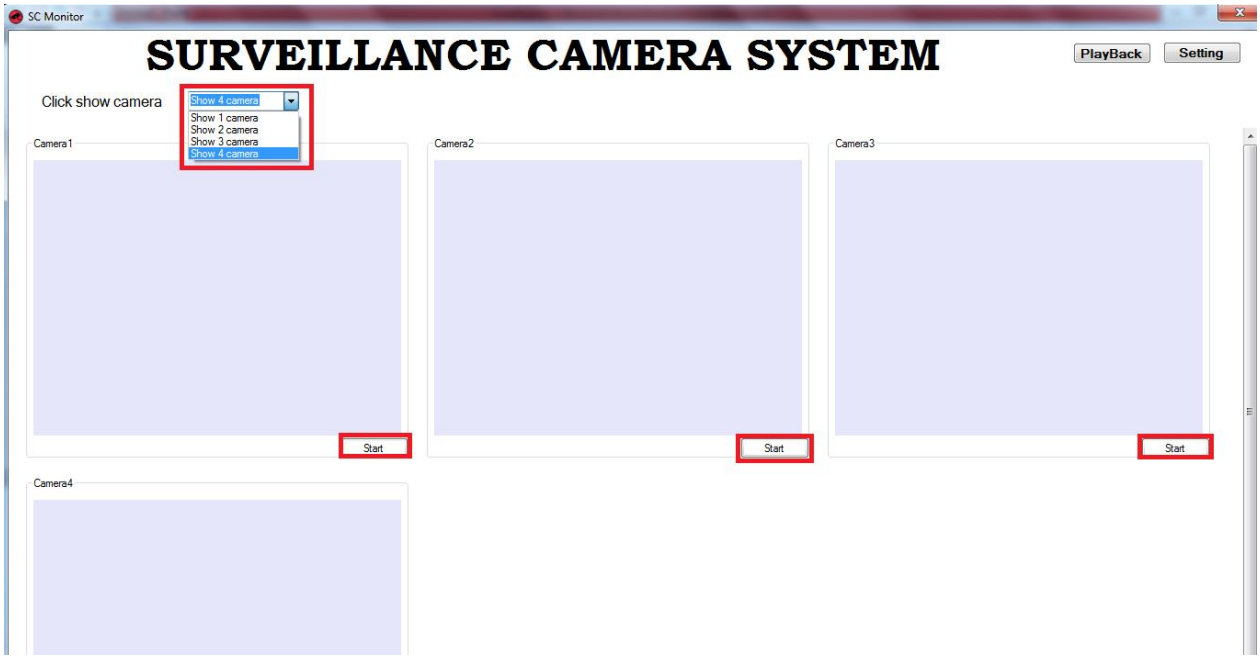

Figure 52 – Before start watching

After clicking on "Start" button, the computer starts receiving video signal from added port number of the camera and the video starts playing in the camera box.

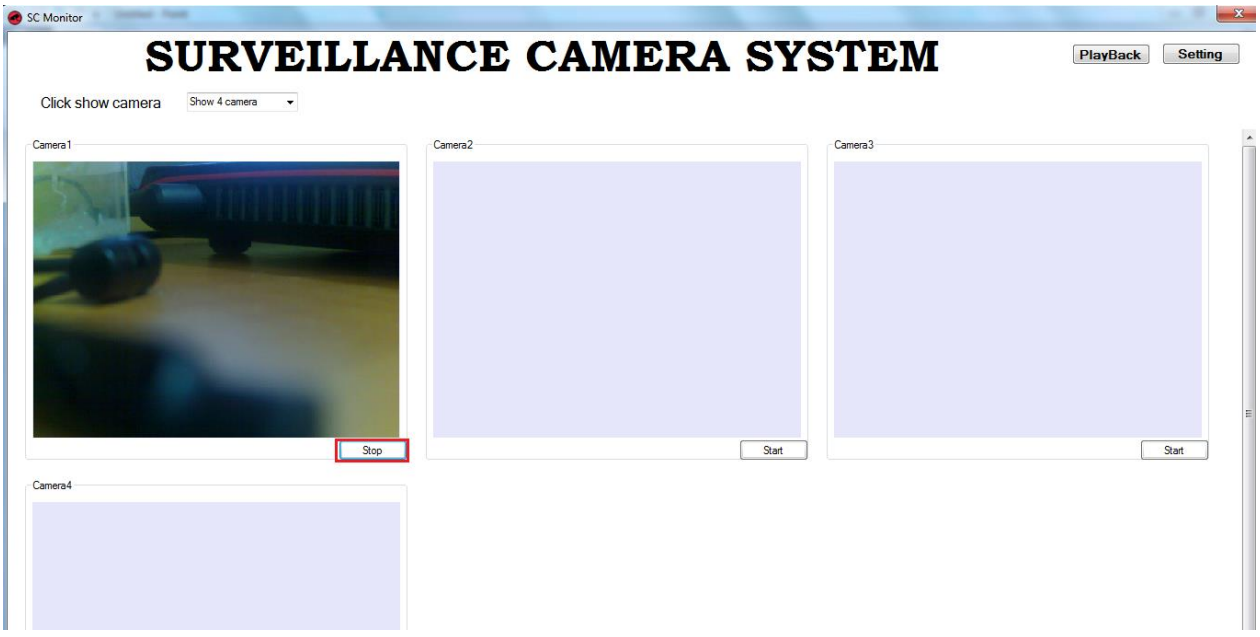

Figure 53 – Start watching

User can click on "Stop" button to stop watching video from a specific camera.

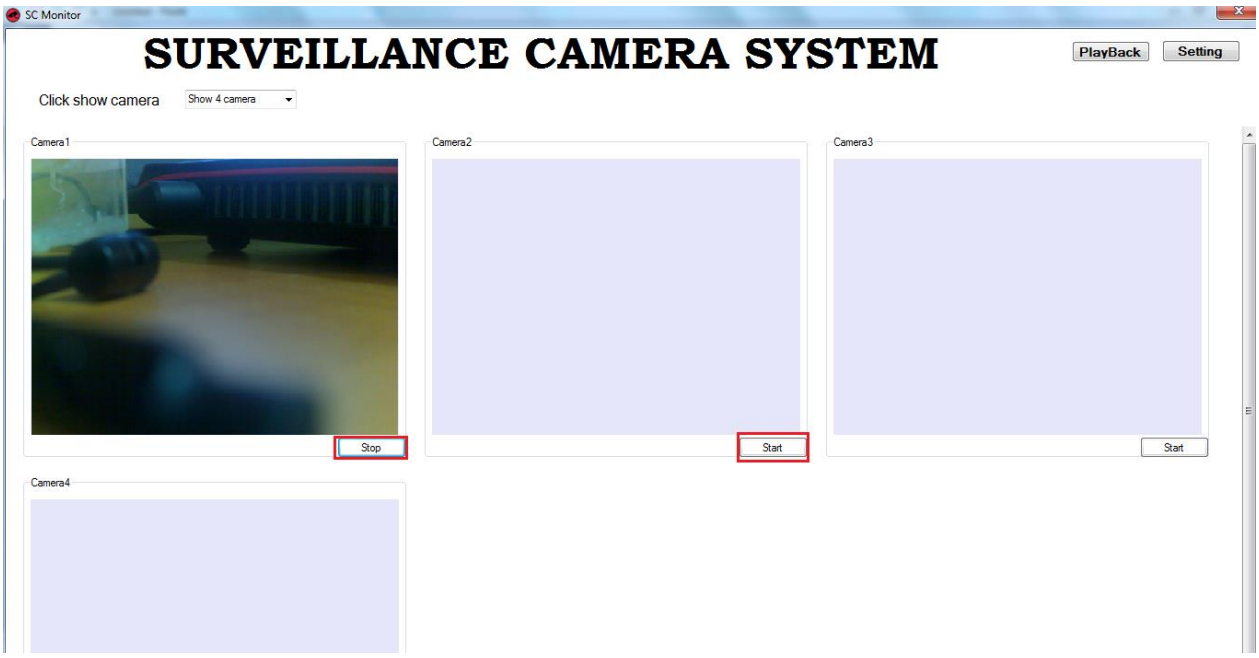

Figure 54 – Stop watching

6.3.2 Control cameras

User can control their cameras by entering setting screen using "Setting" button. The list of camera is shown in setting screen which include all information of cameras (name, port number and description). In this screen, user can add new camera, edit and delete an existing camera.

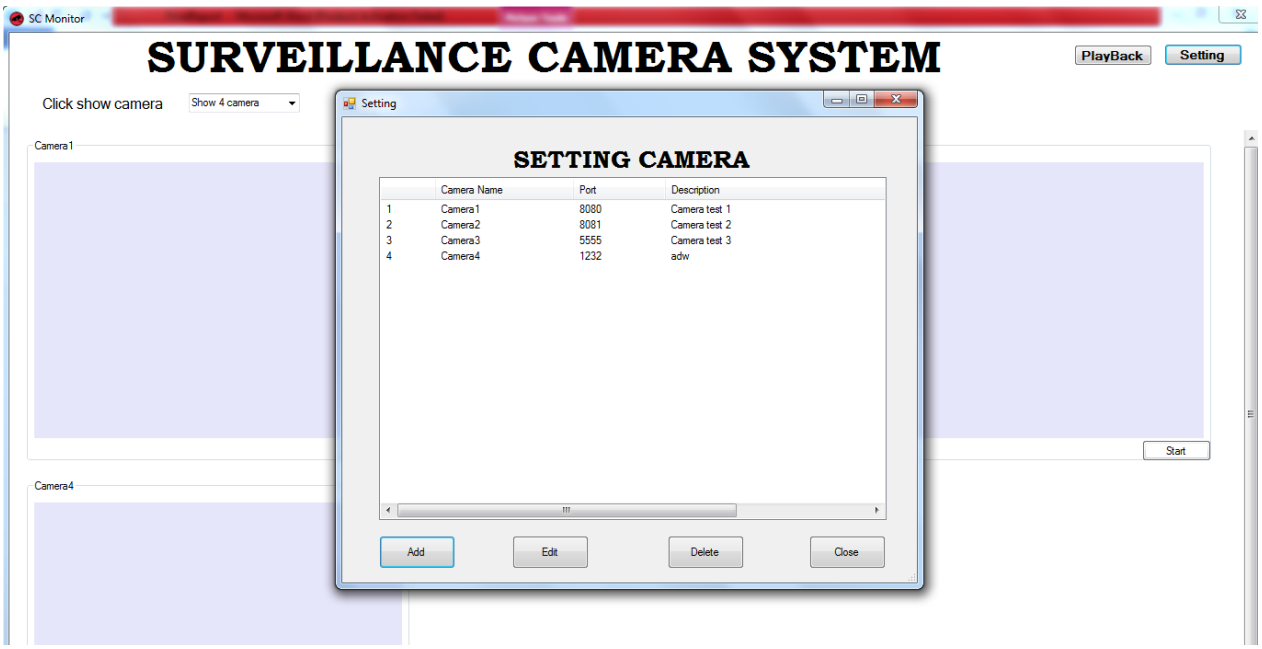

Figure 55 – Setting screen

6.3.2.1 Add camera

Follow these steps below to add a new camera to camera list:

1. Click on "Add" button on the Setting screen.

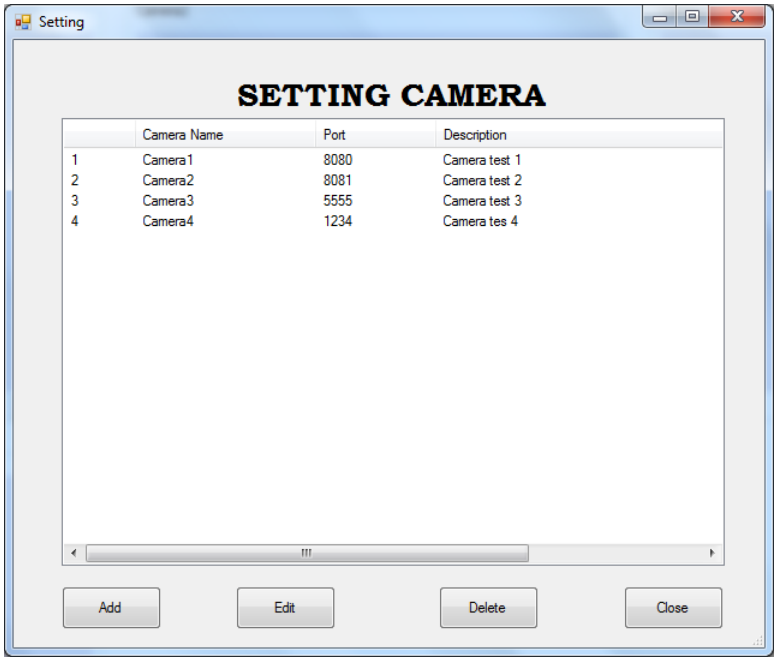

Figure 56 – Choose Add button

2. Adding form appears.

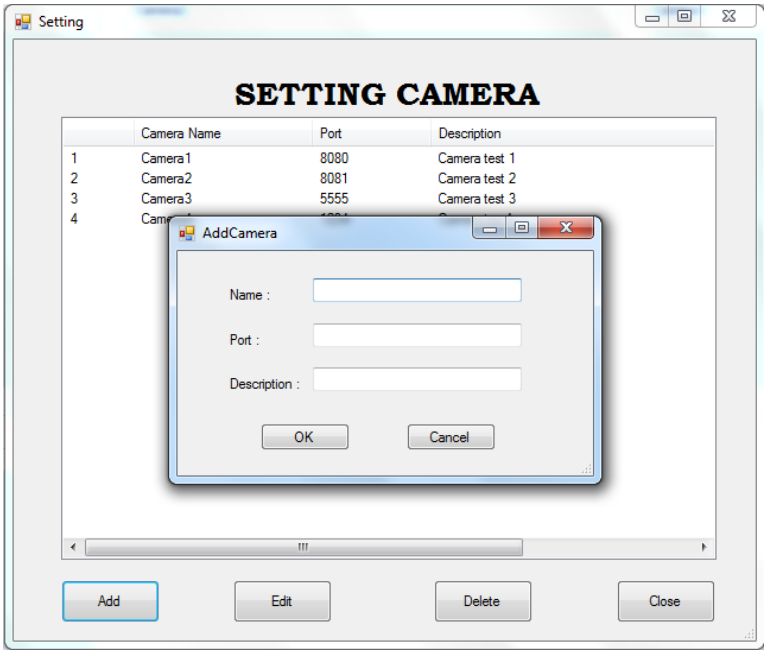

Figure 57 – Adding Screen

3. Fill in textboxes with information of the camera.

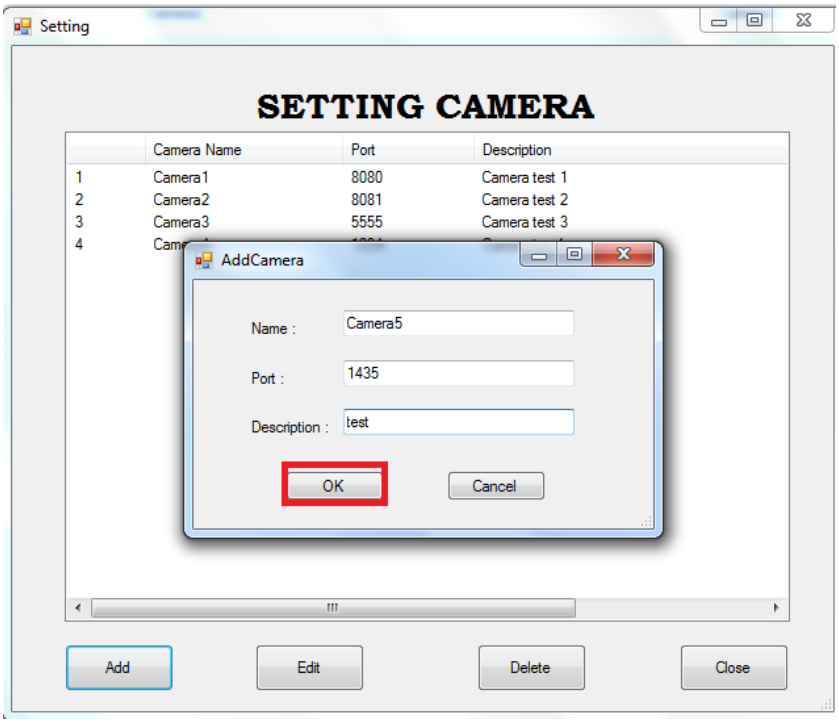

Figure 58 – Enter camera information

4. A message box will appear to announce if the adding is successful. Click on "OK" to close this message box.

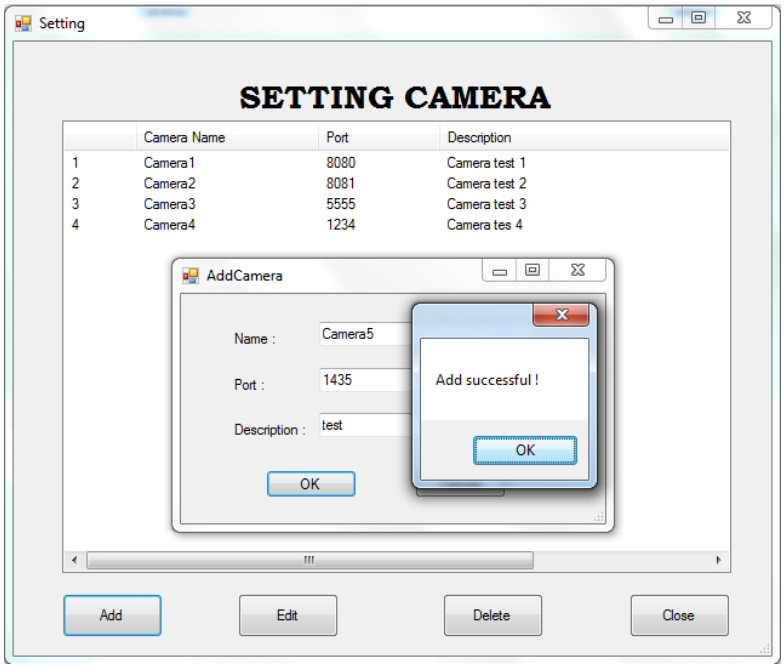

Figure 59 – Add succesful

After above steps, a new camera will be created with entered setting.

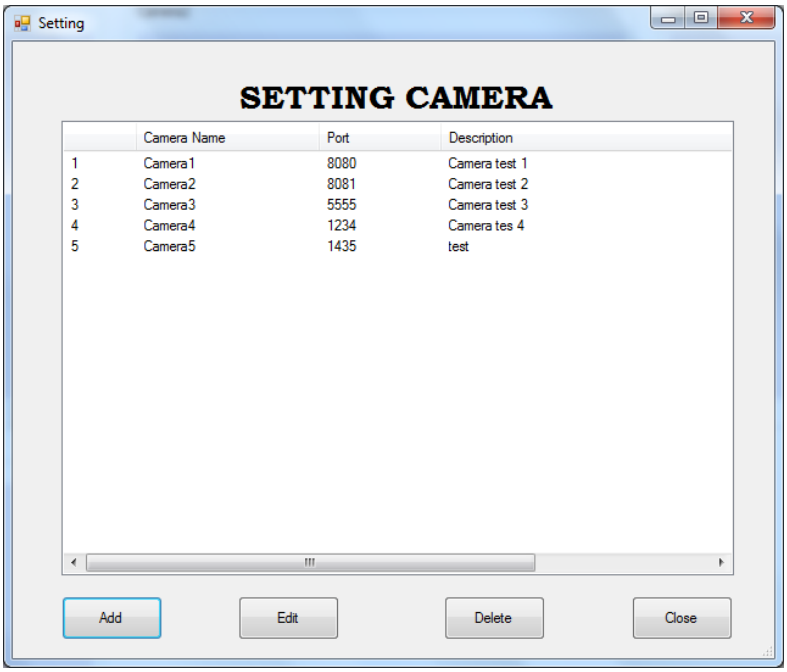

Figure 60 – Return Setting screen

6.3.2.2 Edit camera

Follow these steps below to edit information of an added camera in the camera list:

1. On the Setting screen, choose a camera on the list and click on "Edit" button.

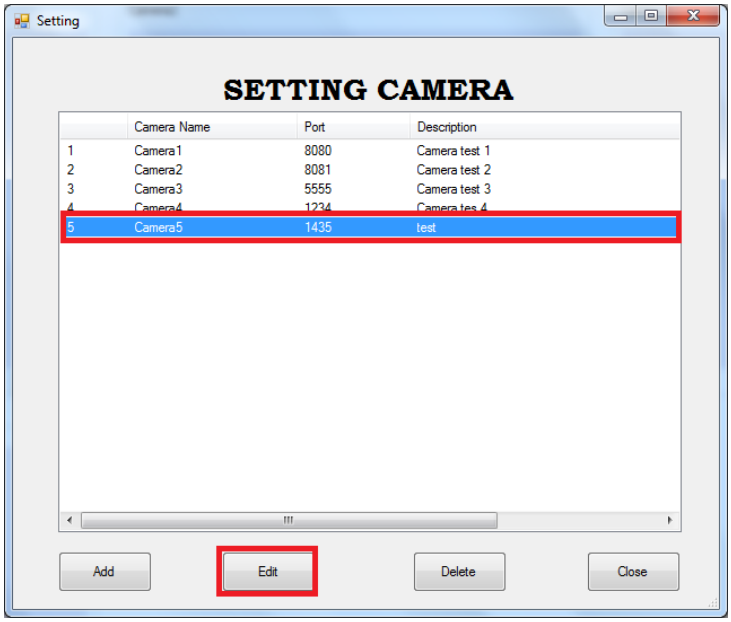

Figure 61 – Choose Camera for editting

2. Camera's information form appear with all information of the chosen camera.

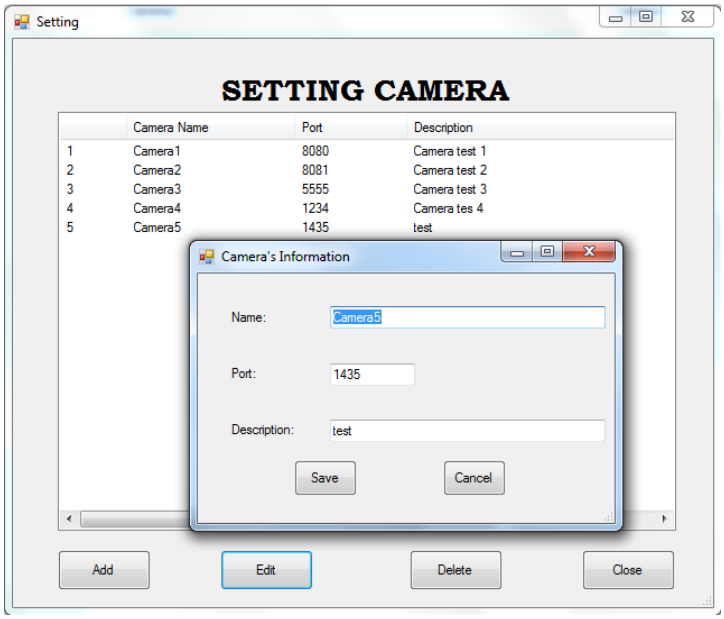

Figure 62 – Editting screen

3. Change information of the camera and click on "Save" button.

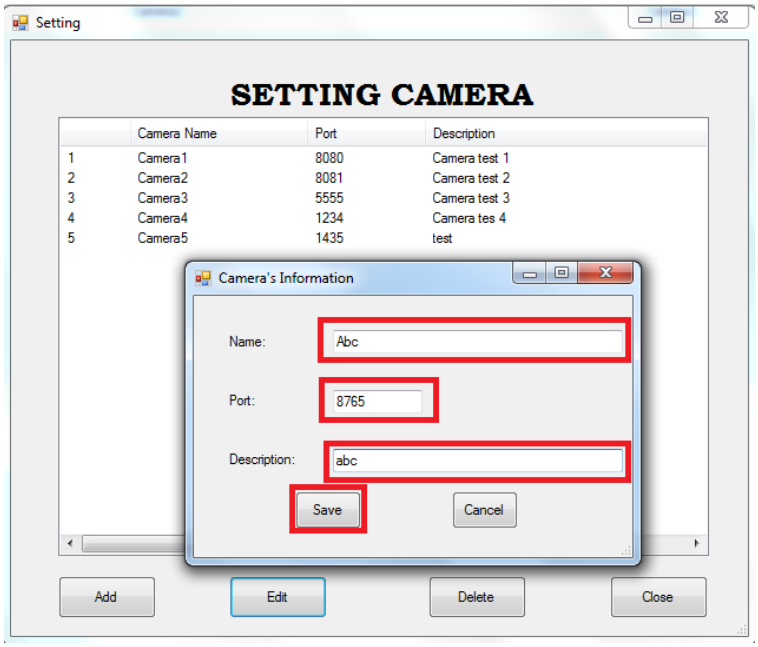

Figure 63 – Change camera information

4. A message box will appear to announce if the updating is successful. Click on "OK" to close this message box.

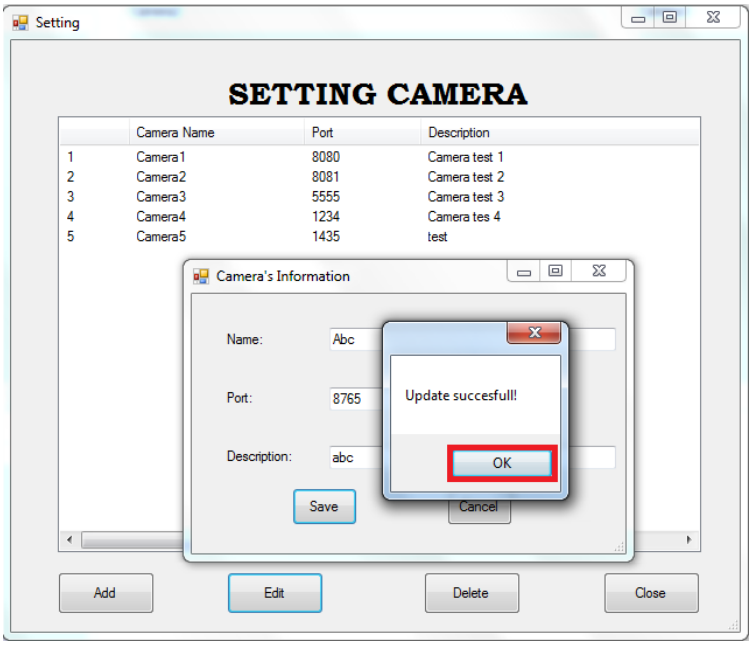

Figure 64 – Edit successful

After above steps, the chosen camera will be updated with new setting.

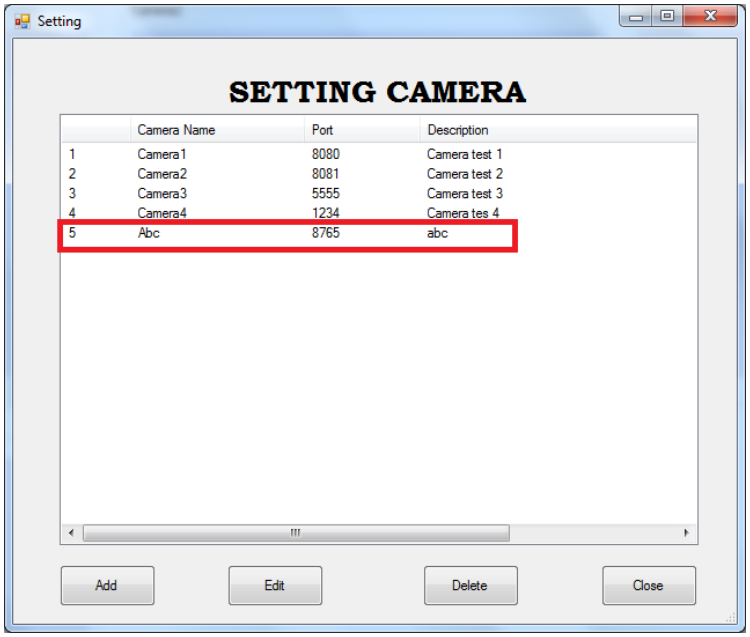

Figure 65 – Return Setting screen

6.3.2.3 Delete camera

Follow these steps below to edit information of an added camera in the camera list:

1. On the Setting screen, choose a camera on the list and click on "Delete" button.

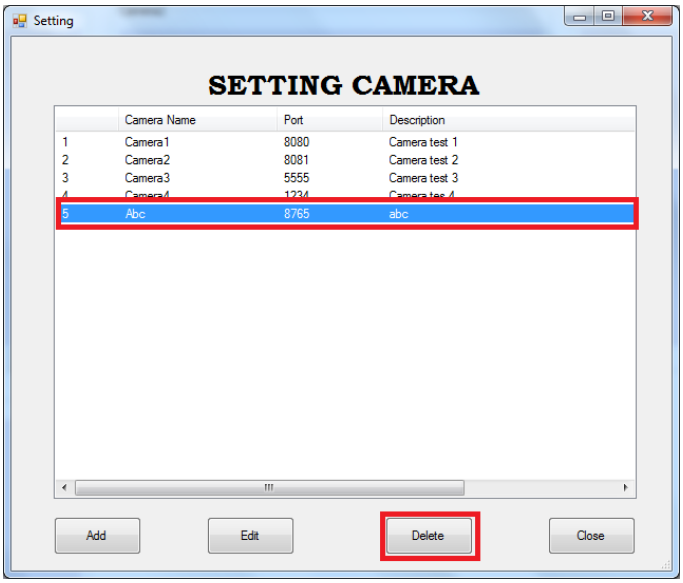

Figure 66 – Choose camera for deleting

2. A dialog appears asking if user is sure want to delete the chosen camera.

Click on "Yes" button.

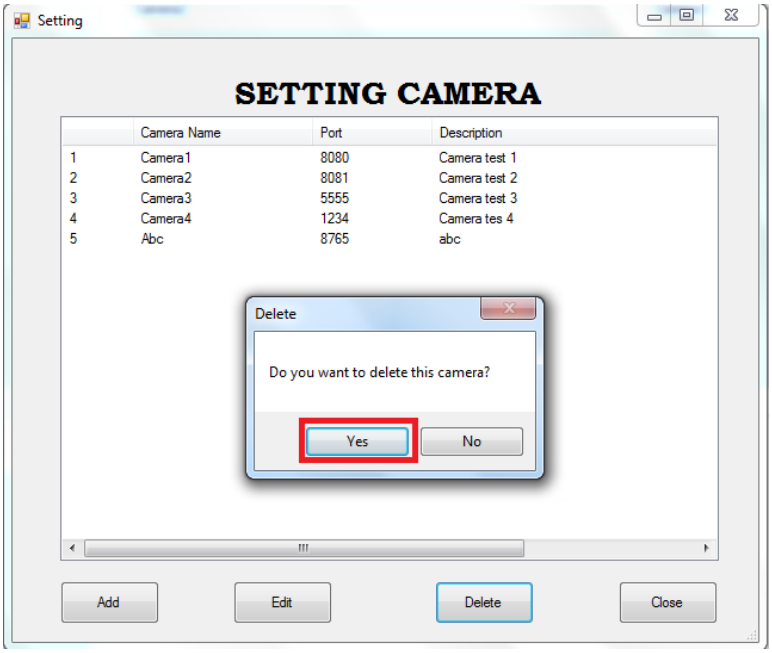

Figure 67 – Delete dialog appear

3. A message box will appear to announce if the deleting is successful. Click on "OK" to close this message box.

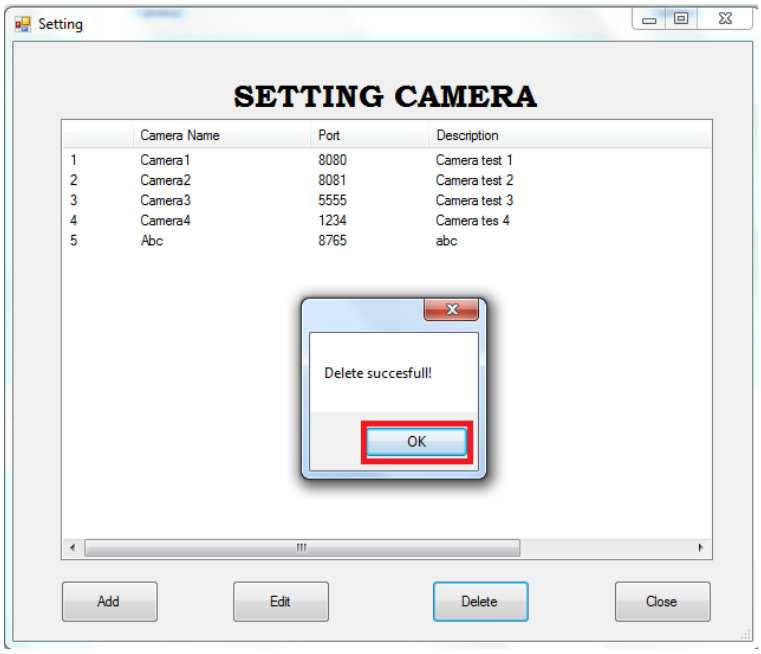

Figure 68 – Delete succesful

After above steps, the chosen camera will be removed from the list.

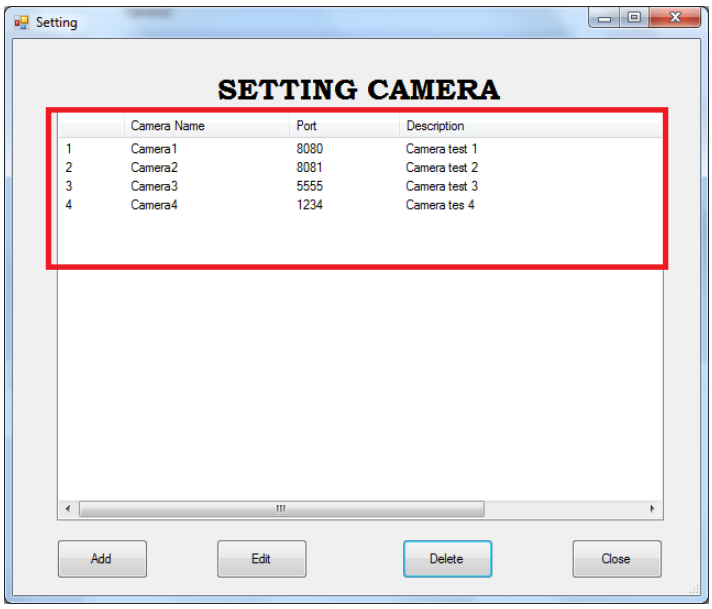

Figure 69 – Return Setting screen

6.3.3 Playback

User can use playback feature by clicking on "Playback" button in the main screen.

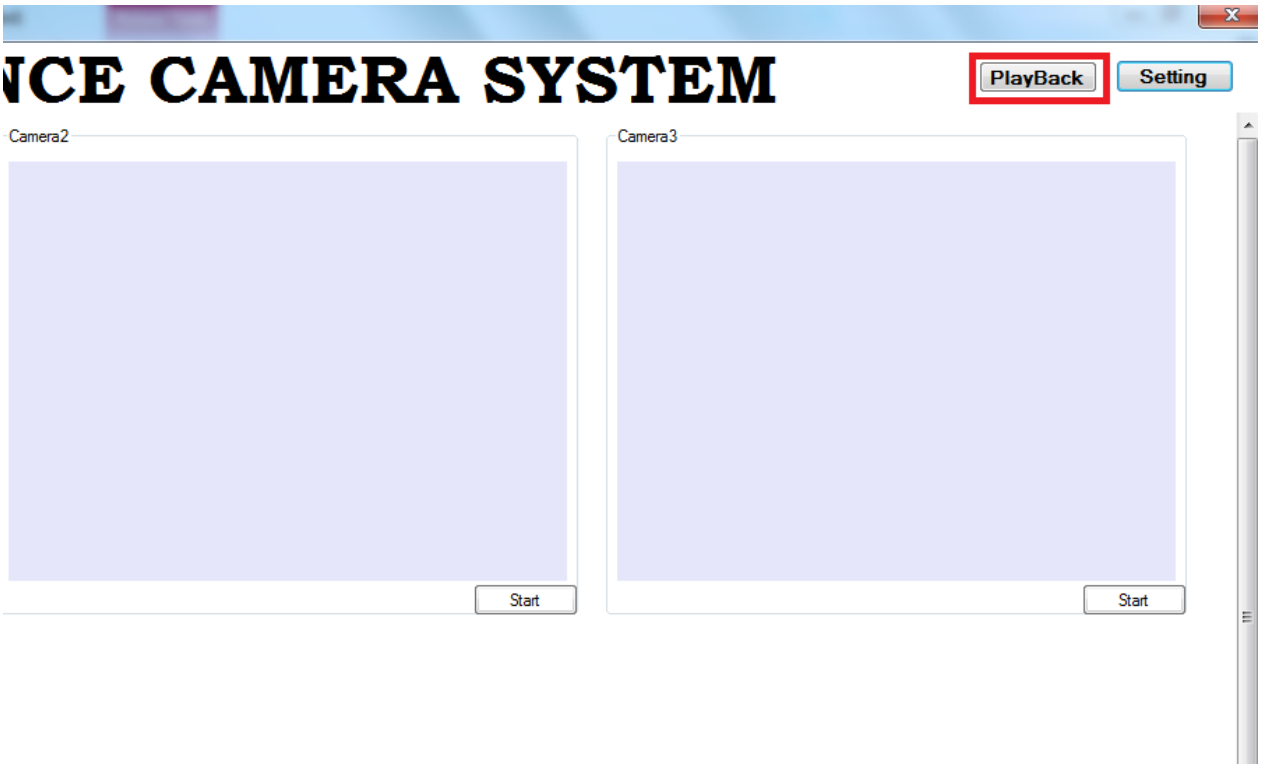

Figure 70 – Choose Playback button

Playback screen will appear, user can choose the date, camera and hour of the

recorded video that they want to watch.

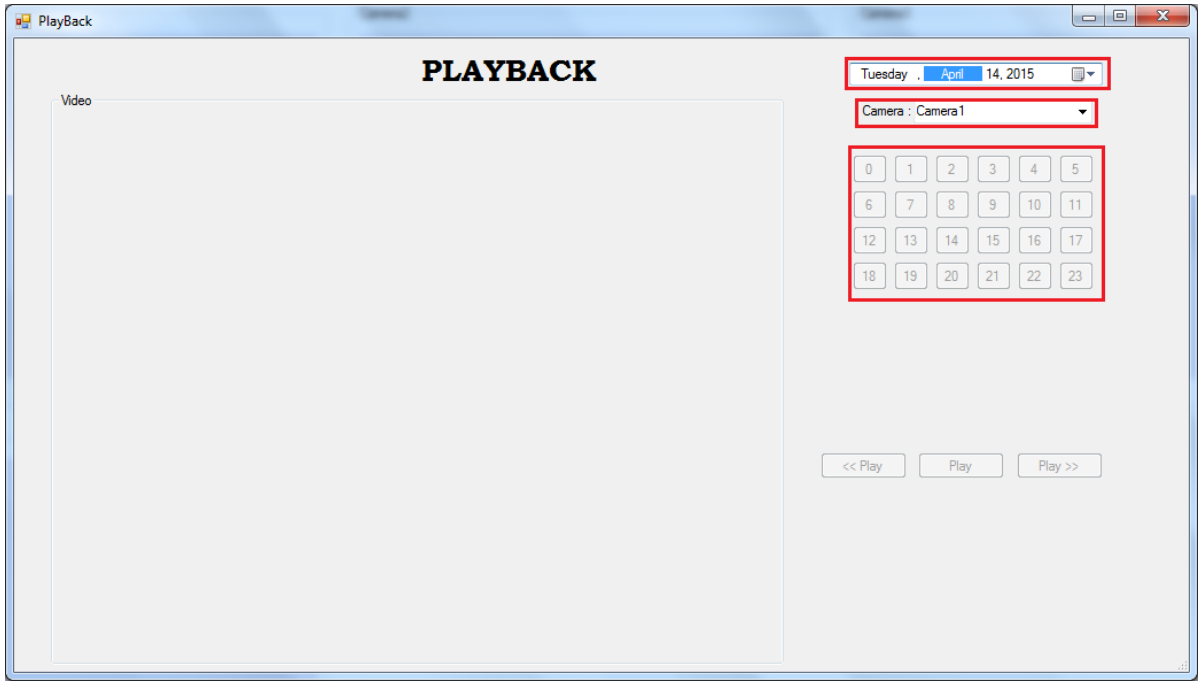

Figure 71 – Playback screen

After choosing camera, hour button will become available if there is videos that were recorded in that time.

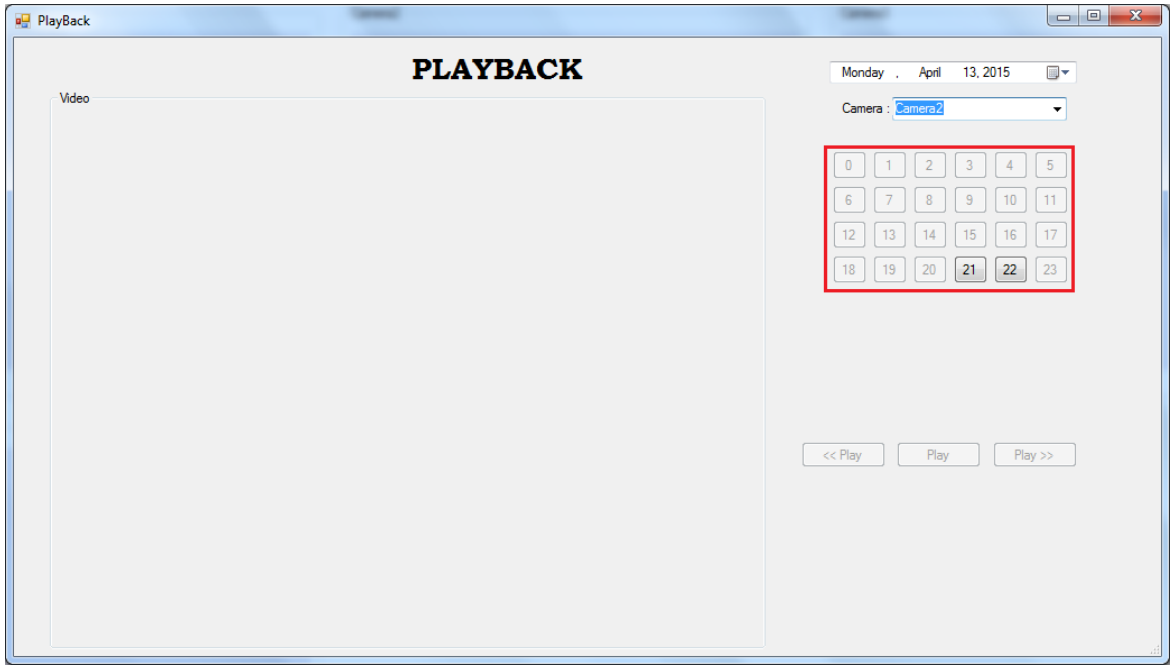

Figure 72 – Hour buttons

Choose an available hour and click on its button, the video will appear in the video

box in the left side. “Play” button will become available.

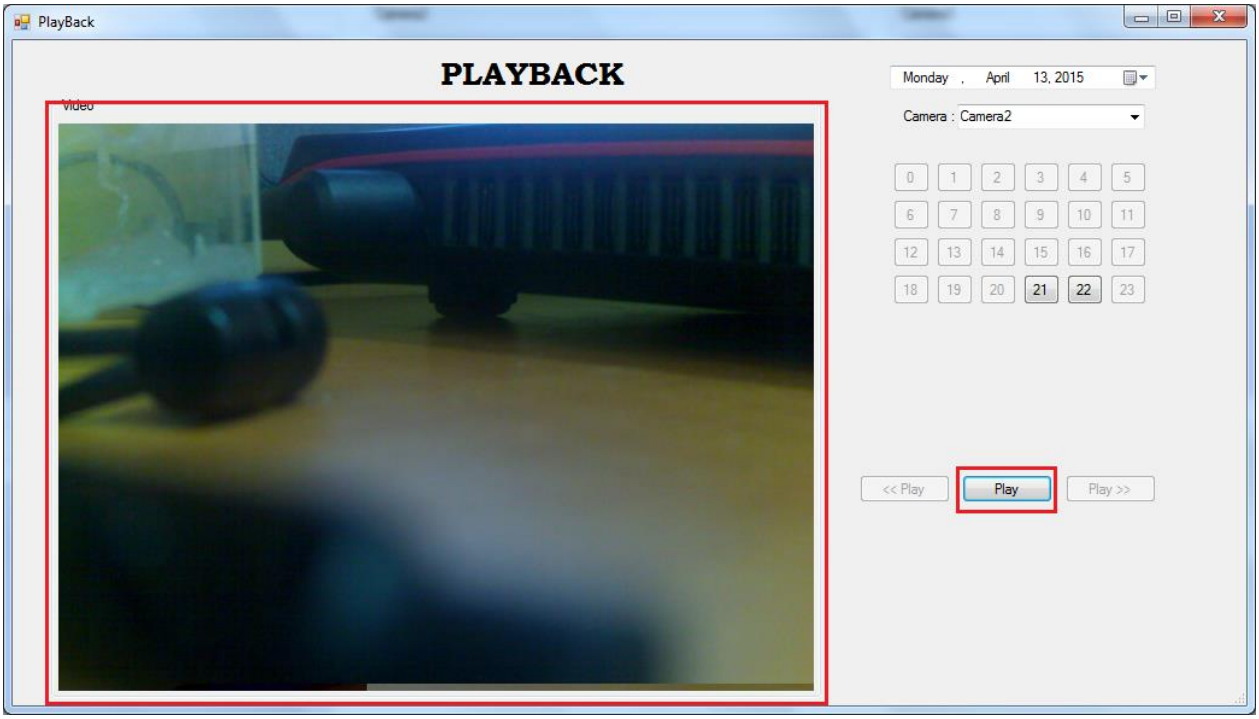

Figure 73 – Video appear

The video starts playing after user click on “Play” button. Two other buttons - “<< Play” and “Play >>”, which help user to play the video with 2X and 4X speed forward or backward, will become available “Play” button also change to “Stop” button.

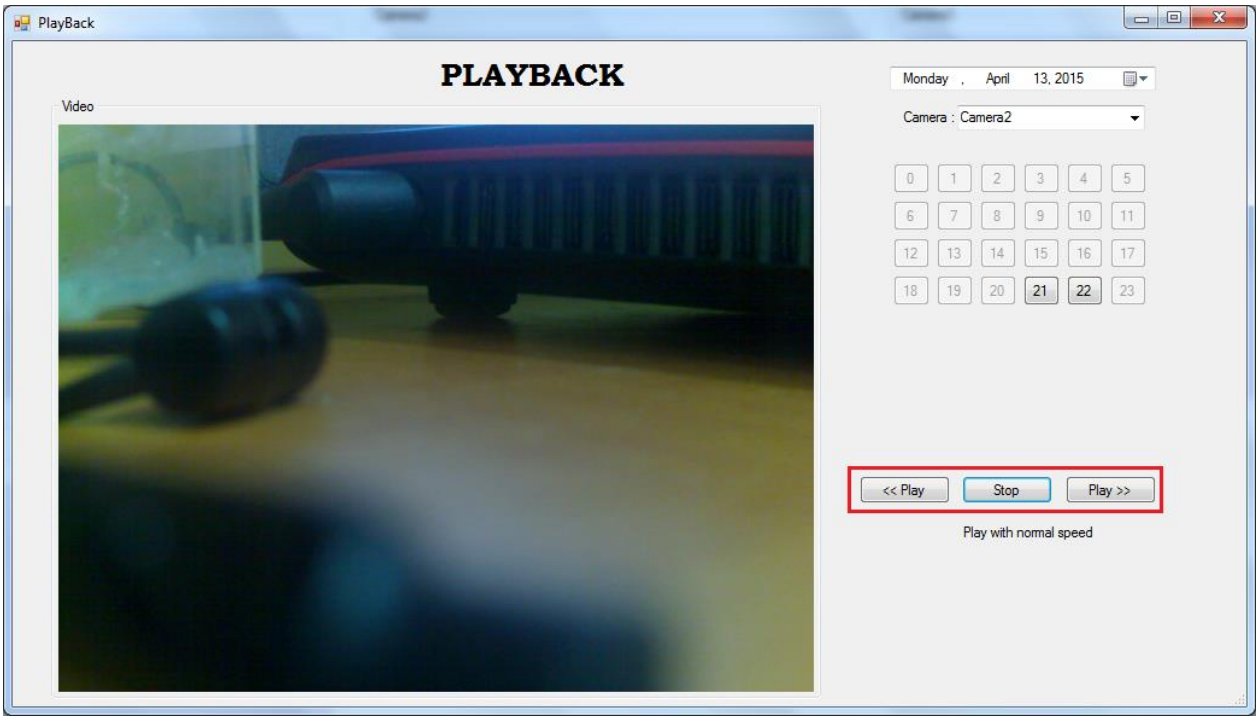

Figure 66 – Play video

7. REFERENCES

- Texas instrument website, version dated (2014/12/7), URL: http://www.ti.com/.
- Holloway, Catherine. JPEG Image Compression: Transformation, Quantization, and Encoding. April 2008. Accessed 22 Sept. 2009.
- Wallace, Gregory K. “The JPEG Still Picture Compression Standard”. Communications of the ACM. 1 April 1991: 30-44. Accessed 4 Sept. 2009.
- Andrew S. Tanenbaum. Computer Networks.
- Douglas E. Comer. Internetworking with TCP/IP - Principles, Protocols and Architecture.
- W. Richard Stevens. The Protocols (TCP/IP Illustrated, Volume 1). Addison-Wesley Professional; 1st edition (December 31, 1993)
- Alexandr Lytkin. IP Video Surveillance. An Essential Guide, 2012.
- Cornett, Ben. "Intro to Surveillance Camera Technologies"
- Poynton, Charles. Digital Video and HDTV: Algorithms and Interfaces.
- Penfield, Paul. “Chapter 3: Compression.” Notes. 12 Feb. 2004. MIT. Accessed 6 Sept. 2009.